Министерство образования и науки Российской Федерации
Федеральное государственное бюджетное образовательное учреждение высшего образования
Московский Государственный Университет им. М.В.Ломоносова
Механико-математический факультет

УДК 519

А.С.Подколзин

Компьютерное моделирование логических процессов

Том 2. Опыт обучения компьютерного решателя задач: логические приемы, алгебра множеств, комбинаторика и элементарная алгебра

Москва, 2015 г.

Деп. №

Организация-депонент: ФГБОУ ВО "Московский Государственный Университет им. М.В.Ломоносова", г. Москва

Название работы: Компьютерное моделирование логических процессов. Том 2. Опыт обучения компьютерного решателя задач: логические приемы, алгебра множеств, комбинаторика и элементарная алгебра.

Автор: Подколзин А.С. (07.12.1950), ФГБОУ ВО "Московский Государственный Университет им. М.В.Ломоносова", г.Москва, Российская Федерация.

Реферат: Книга является вторым томом монографии "Компьютерное моделирование логических процессов", посвященной описанию новой технологии обучения компьютерных решателей задач. Эта технология позволила создать систему компьютерной математики, значительно превосходящую по своим логическим возможностям традиционные системы. Она моделирует рассуждения человека пошаговым образом и позволяет получить не только ответ, но и сам процесс решения. Архитектура логической системы и ее внутренние языки были представлены в первом томе монографии, изданном при поддержке РФФИ в 2008г. Данный том начинает описание опыта обучения решателей. В нем рассматриваются общелогические приемы, а также приемы, относящиеся к следующим разделам: алгебра множеств, простейшие свойства функций, мощности множеств и комбинаторика, числовые множества, элементарная алгебра, комбинаторные функции, многочлены. Текст сопровождается примерами и упражнениями по самостоятельной записи приемов на языках решателя. Описание приемов иллюстрирует широкий спектр способов алгоритмизации теорем, выявленных при анализе более чем 10000 примеров из различных областей, и предоставляет необходимый материал для развития механизмов самообучения компьютерных решателей.

Ключевые слова: Искусственный интеллект, компьютерный решатель задач, компьютерное моделирование, логический процесс.

Язык: рус.

Страниц: 1153

Илл.: нет

Библ: 36

Title: Computer simulation of logical processes. Vol.2.Experience of computer solver teaching: logical rules, set theory, combinatorial theory, elementary algebra.

Author: Podkolzin A.S., Moscow State University (MSU), Moscow, Russian Federation.

Abstract: This book is the second volume of monograph "Computer simulation of logical processes". New technology of computer solvers teaching is described. Based on this technology, computer intellectual system was developed. This system not only gives the answer for problem, but simulates step-by-step the whole process of reasoning. Architecture of logical system and it's languages were presented in first volume. In this volume we observe experience of solver teaching in such areas, as logics, set theory, combinatorial theory and elemenatry algebra. This observation illustrates wide spectrum of theorems algorithmization methods that were extracted from more than 10000 exercises.

Key words: artificial intelligence, computer problem solver, computer simulation, logical process.

Введение

Мечта человечества об искусственном интеллекте имеет давнюю историю. На сегодняшний день она, увы, остается лишь мечтой. Производительность и память современных вычислительных систем огромны и продолжают быстро увеличиваться. Они относительно дешевы и широко распространены. Однако, среднестатистический стандарт использования этих систем по-прежнему сводится к некоему гибриду телефона, записной книжки, пишущей машинки, телевизора и тренажера для отработки простейших реакций. Понимание мира хотя бы на уровне пятиклассника, с его способностью читать книги и воспринимать смысл изображений, остается далеко за рамками возможностей существующих "интеллектуальных систем". О таких вещах, как научно-техническое творчество в подлинном смысле этого слова, говорить и вовсе не приходится. Автоматизировать удается лишь сравнительно простые и узко специализированные функции естественного интеллекта. Так как данное положение наблюдается уже достаточно давно, есть все основания говорить о наличии некоторого принципиального барьера, возникшего на пути к искусственному интеллекту. Очевидно, этот барьер никак не связан с "физическими" возможностями сегодняшних компьютеров, а связан с острым дефицитом наших знаний о логических процессах, лежащих в основе любой интеллектуальной деятельности. По существу, речь должна идти о создании новой науки, изучающей эти процессы, своего рода "логической динамики".

Нельзя сказать, чтобы попыток начать изучение логических процессов не было вовсе. Прежде всего, они предпринимались в рамках математической логики. Однако, ее попытки создания универсальных эффективных процедур решения задач успехом не увенчались. Из общих соображений возникла лишь общая схема организации перебора. Экспоненциально растущая трудоемкость ограничила сферу применимости этих процедур самыми простыми задачами и даже поставила под сомнение возможность создания искусственного интеллекта вообще. Впрочем, человек вполне эффективно решает задачи, избегая "экспоненциальных переборов", и неудача попытки решить проблему математическим путем означает лишь заведомую недостаточность умозрительных построений для анализа такого сложного явления, как интеллект.

С другой стороны, практически сразу после появления компьютеров стали создаваться программы для решения "интеллектуальных" задач в самых различных областях. Они имитировали действия эксперта в соответствующей области и получили название экспертных систем. Рассматривались задачи на доказательство теорем в формальной логике, задачи на доказательство геометрических теорем, формальное интегрирование, тригонометрические уравнения, уравнения в целых числах, шахматные задачи, и многое, многое другое. Собственно, сейчас трудно найти такую предметную область, для которой не было бы создано множество экспертных систем. Некоторые из них весьма развиты и успешно могут конкурировать с человеком. Например, для шахмат была создана система, одержавшая победу даже над чемпионом мира. Может быть, некое объединение всех этих систем и является долгожданным искусственным интеллектом? Однако, при ближайшем рассмотрении становится ясно, что богатство здесь иллюзорное. В сколь-нибудь сложных областях развитие экспертной системы останавливалось на простейших задачах, демонстрируя скорее невозможность освоения области современными средствами, чем успешное ее преодоление. В отдельных случаях (в тех же шахматах) удавалось достичь внешне впечатляющих результатов не за счет проникновения в логику действий человека, а лишь методом "грубой силы", используя огромную производительность компьютеров для перебора. Хорошо известны высказывания на этот счет одного из пионеров в области искусственного интеллекта чемпиона мира по шахматам М.М.Ботвинника.

Разумеется, во многих, сравнительно простых с логической точки зрения, предметных областях были созданы эффективные прикладные экспертные системы. Однако, здесь возникло другое негативное явление. Разработчики, находясь под впечатлением достигнутых успехов, экстраполировали принципы организации своих систем далеко за рамки рассматривавшейся предметной области, возводя их в ранг "общей теории" искусственного интеллекта. Естественно, в сколь-нибудь более сложных или просто сильно отличающихся от исходной предметных областях эти теории никакой пользы не принесли. Здесь прослеживается всё та же линия основанных на недостаточной информации умозрительных построений.

Впрочем, отвлечемся от заведомой слабости какой-то (неважно, малой или большой) части узкоспециализированных экспертных систем. Предположим, что каждая из них в своей области достигла уровня самого опытного эксперта. И в этом случае формальное их объединение никакого искусственного интеллекта не даст. Во-первых, из-за проблем организации взаимодействия. Как известно, знания различных предметных областей тесно взаимосвязаны друг с другом. Нельзя хорошо решать задачи по геометрии, совсем не зная алгебры, нельзя овладеть физикой, не зная математики, и т.д. и т.п. Чтобы научить систему пониманию смысла изображений, нужно заложить в нее примерно столько же знаний о мире и приемов использования этих знаний в рассуждениях, сколько их понадобилось бы для обучения другой системы пониманию естественного языка. Никакой самый умный переключатель между изолированными экспертными системами здесь не поможет, так как решение задач "на стыке" различных областей потребует одновременного их участия. Если бы подобная попытка создания "объединенного" искусственного интеллекта и в самом деле когда-либо была бы предпринята, то она немедленно привела бы к необходимости устранения перегородок между отдельными частями и, фактически, созданию заново некоей "универсальной" экспертной системы.

Но и так мы не получили бы искусственного интеллекта. Экспертная система лишь зафиксировала бы текущий уровень развития наших знаний и умений. Несмотря на свою неоспоримую практическую полезность, она никоим образом не могла бы претендовать на ту роль мощного ускорителя научно-технического прогресса, которую призван сыграть искусственный интеллект. Такая роль однозначно предполагает способность системы к саморазвитию. Простейшая адаптация, связанная с оптимизацией системой каких-то своих параметров, не в счет. Подлинное саморазвитие требует механизмов, создающих новые приемы решения задач на основе имеющихся знаний и пополняющих знания с ориентацией на решение задач. Без этого невозможно решение нестандартных задач, а они в творчестве составляют подавляющую долю. Интеллектуальной системе просто необходима мощная техника компиляции, выводящая ее архитектуру и программы из идей, порождаемых ею же на языке, максимально приближенном к языку теоретических знаний. Собственно генерация идей происходит на пограничном слое между теоретическими знаниями и практическими приемами, который и должен стать главным объектом изучения.

Здесь мы снова приходим к вопросу о целесообразности создания изолированных друг от друга интеллектуальных систем в различных предметных областях. Конечно, для игры в шахматы, знание, скажем, аналитической геометрии может показаться излишним. Однако, механизмы, порождающие новые приемы на основе теоретических знаний, имеют общелогический характер. Каждая предметная область вносит в копилку таких механизмов, типов приемов и стандартов рассуждений что-то свое. И для получения полной коллекции необходимо изучить как можно более широкий спектр различных предметных областей, чтобы уже впоследствии вернуться "во всеоружии" к какой-то одной области.

Таким образом, путь к искусственному интеллекту лежит через создание экспертной системы, охватывающей достаточно широкий спектр предметных областей и программируемой на уровне "пограничного слоя" между теорией и алгоритмами. Она должна послужить своего рода микроскопом, дающим достаточно богатый фактический материал для изучения общих принципов организации логических процессов: принципов эффективного управления рассуждениями при решении задач, принципов извлечения новых приемов из теорем, принципов автоматического развития теорий, и т.д. Альтернативой является лишь продолжение бесплодных попыток умозрительного угадывания этих принципов.

Данная система вовсе не обязана претендовать на роль сколь-нибудь сильного решателя. В первую очередь, она должна давать множество примеров, объясняющих, как можно было бы управлять рассуждениями, чтобы решить ту или иную конкретную задачу, не прибегая к непомерно большому перебору. Разумеется, по мере накопления средств она начнет многое делать самостоятельно. Однако, в иных предметных областях возможность полной проработки темы при обучении "вручную" вообще представляется сомнительной, и здесь придется ограничиться накоплением единичных траекторий процессов, которые все же дадут пищу для обобщений. Действительно сильные решатели должны будут появиться впоследствии, когда накопление знаний о логических процессах позволит создать интеллектуальные системы, способные самообучаться. Пока таких систем нет, и речь идет лишь о начале систематического исследования, опирающегося на некий компьютерный "логический микроскоп".

Предлагаемая монография "Компьютерное моделирование логических процессов" посвящена описанию компьютерной системы, предоставляющей возможности для развития универсальной экспертной системы ("решателя") указанного выше типа и изучения общих принципов организации логических процессов, включая самообучение. В ранее вышедшем первом томе ("Архитектура и языки решателя задач") были представлены общая архитектура системы и используемые при ее обучении языки "ЛОС" и "ГЕНОЛО́Г". Второй из этих языков расположен в точности на пограничном слое между теорией и алгоритмами. Он задает прием как теорему предметной области, снабженную некоторой алгоритмизирующей разметкой. Обучение системы предпринималось в широком спектре предметных областей и позволило выявить достаточно богатую коллекцию способов эффективной алгоритмизации теорем. Использование этой коллекции, с одной стороны, существенно ускорило и упростило процесс "ручного" обучения решателя, а с другой стороны - позволило вплотную приблизиться к автоматическому синтезу приемов. Следует заметить, что не только программирование решателей, но даже традиционное "нелогическое" программирование в конечном счете выводит свои конструкции из теорем. Это означает, что в самообучающихся интеллектуальных системах вообще все программирование неизбежно должно будет проходить через уровень ГЕНОЛО́Га, и он вполне может претендовать на роль "универсального алгоритмического языка будущего". Разумеется, это придает особую значимость развитию компиляторов для языков такого типа. При развитии ГЕНОЛО́Га и обучении решателя, происходивших одновременно, были рассмотрены такие предметные области, как элементы дискретной математики, алгебра множеств, элементарная алгебра, элементарная геометрия, аналитическая геометрия, линейная алгебра, математический анализ, дифференциальные уравнения, комплексный анализ, вычисления, теория вероятностей, элементы общей алгебры, ряд разделов элементарных физики и химии, шахматы, текстовые задачи (в двух вариантах - на логическом либо естественном языке), анализ рисунков. Обучающий материал содержал примерно 11000 задач, из которых были извлечены примерно 39000 приемов. В отдельных областях даже такое, сугубо предварительное, обучение позволило выйти на неплохой уровень решения задач средней сложности. В других - удалось проработать лишь небольшое количество примеров, которые, однако, дали возможность существенно скорректировать первоначальные представления и получить сравнительно устойчивую архитектуру для дальнейшего развития.

Предпринимается анализ возможностей самообучения системы. К явлению самообучения следует относиться с определенной осторожностью, так как пока известна лишь одна самообучающаяся в абсолютном смысле интеллектуальная система - это вся человеческая цивилизация в целом. Процессы подобного масштаба далеко выходят за рамки возможностей современных компьютеров. Практический интерес представляет сейчас "ограниченное" самообучение отдельного человека, заключающееся в самостоятельном усвоении по книгам или иным источникам возможно большей суммы знаний и приобретении навыков их практического использования путем тренировки на обучающих примерах. Целью работы с компьютерной системой должно явиться освоение аналогичного режима самообучения: ей сообщается теория и дается поток задач, на котором происходят самостоятельное создание приемов и оптимизация их совместного поведения. Центральным здесь является создание развитой классификации логических типов приемов, которая позволяла бы по заданной теореме генерировать серию предположительно интересных для задачи приемов ("идей"), проверять их ценность и отбирать по окончании решения задачи те приемы, которые оказались результативными. Работа над созданием такой классификации начата, однако она связана с трудоемкой итеративной стандартизацией приемов созданного (в общем, пока относительно "сырого") решателя и оптимизацией его логических режимов.

Подробнее о механизмах самообучения, развивающихся в предлагаемой системе, будет рассказано в последнем, пятом томе монографии. Эти механизмы представляют собой некоторую надстройку над решателем и подсказаны его содержимым. Решатель здесь сам начинает играть роль "задачника", поставляющего материал для ручного обучения указанных механизмов. Они достаточно разнообразны и, по всей вероятности, потребуют для своего автоматического развития логических процессов еще более высоких уровней. В целом, складывается перспектива постепенного прохождения "снизу вверх" через некоторую пирамиду уровней логической автоматики, в которой каждый следующий уровень обеспечивает саморазвитие предыдущего и выводится из него путем ручного обучения. В этом смысле, приходится констатировать, что "обучение" и "самообучение" в исследовании логических процессов тесно взаимосвязаны.

Многообразие приемов решателя, возникших при его обучении, описывается во втором, третьем и четвертом томах монографии ("Опыт обучения компьютерного решателя задач"). Данный, второй, том начинает этот цикл. В нем рассматриваются общелогические приемы, а также приемы, относящиеся к следующим разделам: алгебра множеств, простейшие свойства функций, мощности множеств и комбинаторика, числовые множества, элементарная алгебра, комбинаторные функции, многочлены. В третьем томе будут рассмотрены приемы, относящиеся к математическому анализу, дифференциальным уравнениям и элементарной геометрии. В четвертый том войдет описание приемов по аналитической геометрии, линейной алгебре, комплексному анализу, теории вероятностей, общей алгебре, дискретной математике, программированию нелогических вычислений, элементарным физике и химии, шахматам, анализу текстов естественного языка и изображений.

Прежде всего, эти тома следует рассматривать как учебное пособие по программированию на ГЕНОЛО́Ге. В различных предметных областях рассыпаны многочисленные "особые случаи", для которых приходилось пополнять копилку возможностей ГЕНОЛО́Га. Они могут многое подсказать при обучении решателя в новых областях. Текст книги хорошо соответствует оглавлению приемов, имеющемуся в системе, и читать ее следует, имея перед глазами описание текущего приема на экране компьютера. Первые разделы книги сопровождаются примерами и упражнениями. В свете приведенных выше соображений, полное описание приемов решателя необходимо не только как справочник по ГЕНОЛО́Гу, но и как отправная точка для последующей оптимизации логической автоматики и развития общих механизмов самообучения. В сочетании с компьютерной системой, оно делает видимыми многие проблемы, которые еще предстоит решать.

Параллельно с написанием монографии продолжается развитие системы. Поэтому каждая книга будет сопровождаться своей версией программы. Все эти версии, расположенные в хронологическом порядке, могут быть получены по адресу " www.intsys.msu.ru/invest/solver/logsyst.zip". Сборники задач, использованные при обучении решателя, приводятся в списке литературы.

Автор выражает искреннюю благодарность В.Б.Кудрявцеву, поддержка которого сделала возможным проведение данного исследования. Автор благодарен также П.А. Пантелееву, оказавшему помощь при подготовке рукописи.

Оглавление

Глава 1

Общелогические приемы, реализованные на ЛОСе

Для обучения и оптимизации решателя, а тем более для извлечения из него информации о возможных подходах к самообучению, необходимо хотя бы общее представление о тех приемах, которые уже имеются, а также о выработанных стандартах технической обработки приемов. Данный том посвящен систематическому описанию накопленного в решателе многообразия приемов, которое одновременно может играть как роль справочника по решателю, так и роль учебника по программированию решателей. Не все предлагаемые технические решения бесспорны, и описание их во многих случаях следует рассматривать лишь как постановку задачи на поиск более точного объяснения. Не следует забывать, что предлагаемая версия решателя лишь открывает поле исследований по логической динамике, представляя собой что-то вроде чернового наброска, иллюстрирующего созданные инструментальные возможности. Предстоит очень большая работа по оптимизации имеющихся конструктивных решений, в процессе которой практически все содержимое, вероятно, будет полностью переделано. С другой стороны, созданные процедуры, все же, работают на многих тысячах задач, архитектура их подсказана обучающим материалом, и любые крупные изменения архитектуры должны быть хорошо с этим материалом согласованы. Степени свободы по варьированию решателя могут оказаться во многих случаях не такими уж и большими.

Описание базы приемов мы начнем с общелогических приемов, применение которых возможно в произвольных предметных областях. Подавляющее большинство приемов решателя реализованы на ГЕНОЛОГе. В принципе, возможно такое развитие этого языка, при котором все приемы были бы переведены на него. Однако, в настоящее время часть приемов реализованы непосредственно на ЛОСе. Главным образом, это общелогические приемы, где оказалось проще написать программу вручную, чем создавать новые компоненты ГЕНОЛОГа для почти одноразового применения. Ниже перечисляются основные реализованные на ЛОСе общелогические приемы, которые могут быть найдены в разделе "Приемы решателя" - "Общие приемы" оглавления программ. Технические подробности программ приемов опускаются. Эти программы легко находятся в концевых пунктах указанного раздела, так как названия пунктов примерно соответствуют подзаголовкам данного текста.

1.1 Приемы задач на описание

Попытка применения приема обычно начинается с усмотрения в задаче некоторого ключевого для данного приема логического символа. Такой принцип позволяет равномерно распределить приемы по различным логическим символам и обеспечить отсечение всех приемов, не связанных с контекстом задачи. В процессе обучения выяснилось, что число приемов, применение которых невозможно связать с каким-либо ключевым понятием, возникающим в задаче, крайне невелико. Эти приемы распределены по четырем типам задач - на описание, преобразование, доказательство и исследование. Попытка применения приема начинается здесь с рассмотрения логического символа, являющегося типом задачи, и обращения к программе этого символа.

В данном подразделе перечислим приемы, отнесенные к задачам на описание. Их программы собраны в ветви фрагментов программ логического символа "описать". Переходить к начальным точкам программ отдельных приемов следует через концевые пункты оглавления программ.

Учет о.д.з.

Для сокращения формулировок задач принимаются специальные соглашения об опускании условий, восстанавливаемых в процессе решения по умолчанию. Так, обычно опускаются сопровождающие условия, обеспечивающие осмысленность выражений и утверждений основных условий. Эти условия называются условиями на область допустимых значений; сокращенно - о.д.з. Чаще всего с условиями на о.д.з. приходится сталкиваться в задачах по элементарной алгебре. Например, если решается уравнение

$$\frac{2}{2+\sqrt{4-x^2}} - \frac{1}{2-\sqrt{4-x^2}} = \frac{1}{x},$$

то перед началом преобразований восстанавливаются условия, выражающие неотрицательность выражений под радикалами и отличие знаменателей от 0: $0 \leq 4-x^2$, $2-\sqrt{4-x^2} \neq 0$, $x \neq 0$. К этой же категории восстанавливаемых по умолчанию условий относится и утверждение "x- число".

Чтобы решатель автоматически присоединял к условиям и посылкам задачи утверждения, задающие о.д.з., имеется специальный прием, точнее, по одному приему для задач каждого из четырех типов. Они устроены совершенно одинаково - проверяется наличие у задачи цели "одз"; если эта цель есть, то она удаляется (для предотвращения повторных применений приема), и происходит обращение к вспомогательной процедуре "одз", входным данным которой является текущая задача. Последняя процедура и преобразует задачу, пополняя условия и посылки. Заметим, что цель "одз" присутствует в целевых установках почти всех задач, создаваемых в задачнике. В случае задачи на доказательство, не имеющей никаких целей, символ "одз" регистрируется в комментариях, где он обрабатывается аналогичным образом.

Использовать цель "одз" во вспомогательных задачах, создаваемых приемами решателя, следует достаточно осторожно, так как ввод дополнительных условий на о.д.з., строго говоря, не является эквивалентным логическим переходом. Это, скорее, часть интерфейса системы. Однако, в некоторых случаях решатель может сам отбрасывать имеющиеся у него сопровождающие условия на о.д.з., руководствуясь теми или иными принципами восстановления их "по умолчанию". Например, это происходит при выдаче ответа. Лишь для такого восстановления и нужно применять обращения к вспомогательным задачам, имеющим цель "одз".

Информацию, необходимую для определения условий на область допустимых значений, процедура "одз" получает при помощи справочника "одз". Этот справочник имеет своими входными данными следующие значения: x1 - задача; x2,x3,x4 - координата вхождения в нее первого символа выражения или утверждения A. Символом обращения к справочнику служит заголовок терма A. Результатом обращения является набор условий, при которых A "имеет смысл". В особых случаях приемы справочника реализованы на ЛОСе, но основная их часть задана на ГЕНОЛОГе. Теорема приема справочника "одз" имеет вид "длялюбого($x_1 \dots x_n$ эквивалентно(определено($f\ x_1\ \dots\ x_n$)и($B_1 \dots B_m$)))", где $B_1, \dots, B_m$ - условия, при которых $f(x_1 \dots x_m)$ является осмысленным утверждением либо выражением. Например, в случае дроби имеем теорему: "длялюбого($x_1\ x_2$ эквивалентно(определено(дробь $x_1\ x_2$)и(число(x_1)число(x_2)не(равно(x_2 0)))))".

Отметим, что на той же самой теореме приема основан еще один справочник - "типданных". Этот справочник определяет допустимый тип значений операндов операции либо отношения. С его помощью решатель автоматически присоединяет к задаче указания на типы значений переменных и сохраняет их в задачнике еще до обращения к решению задачи - сразу же по окончании ее ручного ввода. Поэтому, хотя справочник "одз" и указывает на типы значений операндов, они обычно уже имеются в наличии и процедурой "одз" к задаче не добавляются.

Приведем краткое описание действий, выполняемых процедурой "одз". Сначала просматриваются все условия и посылки задачи, и для каждого вхождения v символа операции предпринимается обращение к справочнику "одз". Если справочник определяет непустой список утверждений, выражающих условия на область допустимых значений операндов данной операции, то просматриваются не являющиеся посылками задачи элементы F этого списка. Отбрасываются утверждения без свободных переменных, а также утверждения вида $P(g(\dots))$, где P - название такого типа объектов, что операция g принимает только значения этого типа. Принимается решение, рассматривать ли F как новую посылку (условие) или размещать его внутри того же терма, к которому относится v. Последнее делается, если v расположено внутри квантора, описателя или дизъюнкции. Для квантора общности принимается решение об отнесении F к консеквенту либо антецеденту. По итогам просмотра условий и посылок задачи составляется список S четверок ($A_1\ A_2\ A_3\ A_4$), где A_1 - указатель посылки либо условия (0 или 1); A_2 - указатель регистрации сопровождающих условий в качестве новых термов задачи (0) либо вхождение точки регистрации их в старый терм; A_3 - вхождение того терма задачи, к которому относится сопровождаемое вхождение v; A_4 - список сопровождающих утверждений, регистрируемых согласно адресу A_1, A_2.

Список S играет роль плана регистрации в задаче новых утверждений, обеспечивающих сопровождение по о.д.з. Перед реализацией данного плана выполняется упрощение найденных утверждений. Составляющие список четверки ($A_1\ A_2\ A_3\ A_4$) последовательно просматриваются; сначала - относящиеся к посылкам, затем - к условиям. В каждом из случаев просмотр начинается с четверки, относящейся к утверждениям, регистриуемым непосредственно в списке посылок либо условий. Такое упорядочение возникло с целью подготовить к нужному моменту те сопровождающие утверждения, которые понадобятся для упрощения других сопровождающих утверждений. Далее составляется список P тех посылок или условий задачи, которые образуют контекст, относительно которого рассматриваются все утверждения списка A_4.

Чтобы упрощать относительно P указанные утверждения, применяется нормализатор "нормодз". Этот нормализатор реализован на ГЕНОЛОГе; его можно найти в разделе "Логические приемы" - "Общие приемы" оглавления базы приемов. Здесь собраны самые простые упрощающие преобразования типичных условий на о.д.з. - например, преобразование отрицания равенства нулю произведения в конъюнкцию отрицаний равенства нулю сомножителей; замена отрицания равенства нулю степени на отрицание равенства нулю ее основания; замена нестрогого неравенства на строгое при наличии в посылках отрицания равенства; отбрасывание заведомо строго положительных либо строго отрицательных сомножителей в условиях положительности либо отрицательности (разумеется, с необходимой перестановкой операндов), и т.п. Необходимость упрощать сопровождающие условия до регистрации их в задаче, где они все равно были бы упрощены, продиктована соображениями оптимизации. Цепочка мелких преобразований, выполняемых при сканировании задачи, требует обычно в десятки раз больше времени, чем реализация этих же преобразований за один шаг какой-либо вспомогательной процедурой, применяемой вне сканирования задачи. В данном случае имеется хороший повод сгруппировать приемы, упрощающие условия на о.д.з., в одном пакете, так как неупрощенная версия условий обычно порождает длинные цепочки применений этих приемов, а число самих приемов невелико.

Сопровождающие утверждения, регистрируемые в заданной точке, сначала упрощаются относительно контекста P независимо друг от друга. После этого реализуется цикл повторных упрощений, где контекстом преобразований каждого сопровождающего утверждения служит список P, пополненный остальными сопровождающими утверждениями. Преобразования выполняются до тех пор, пока происходят какие-либо изменения. После этого измененный список регистрируется в качестве элемента A_4 текущей просматриваемой четверки.

По завершении упрощения сопровождающих утверждений предпринимается регистрация их в задаче. Сначала рассматриваются те утверждения, которые размещаются во внутренних логических контекстах уже имеющихся посылок либо условий, затем - утверждения, регистрируемые непосредственно в виде новых условий или посылок.

Программа процедуры "одз" может быть найдена в разделе оглавления программ "Приемы решателя" - "Общие процедуры, используемые в приемах" - "Процедура ОДЗ". Программная переменная x2 играет роль накопителя списка S. Программу приема, обеспечивающего учет о.д.з. для задач на описание, можно найти в разделе "Приемы решателя" - "Общие приемы" - "Задачи на описание" - "Учет о.д.з. в задаче на описание". Этот прием применяется на нулевом уровне сканирования.

Создание комментариев, определяющих схему сопровождения по о.д.з.

В процессе решения задачи могут возникать новые выражения, для которых условия на о.д.з. не будут непосредственно усматриваться из контекста так, как они усматриваются для исходных выражений после применения предыдущего приема. Вместе с тем, условия на о.д.з. часто проверяются различными приемами перед выполнением преобразований. Чтобы из-за невозможности быстро проверить эти условия не заблокировать необходимые приемы, в решателе предусмотрена автоматическая коррекция сопровождения по о.д.з. Те немногие основные процедуры, которые фактически преобразуют задачу (например, процедура "заменавхождения"), определяют условия на о.д.з. для новых выражений, проверяют их и регистрируют в контексте задачи. Чтобы облегчить им работу, иногда в приеме содержатся прямые указания на вывод утверждений, сопровождающих по о.д.з. новые выражения. Однако, сопровождающие по о.д.з. утверждения нужно защищать от применения к ним каких-бы то ни было, даже самых простых, преобразований - ведь любое такое преобразование увеличивает "расстояние" между сопровождающим утверждением и сопровождаемым термом, что приводит к замедлению проверки по о.д.з. или даже к отказу при этой проверке. Для такой защиты предусмотрен специальный комментарий (сопровождение A). В случае задачи на исследование он является комментарием к списку посылок, иначе - к самой задаче. В наборе A данного комментария перечисляются всевозможные пары $(M\ t)$, где t - выражение, для которого необходимо сопровождение по о.д.з.; M - множество утверждений, из которых следует выполнение условий на о.д.з. для t. Предполагается, что эти утверждения встречаются в задаче в явном виде - как посылки, условия, конъюнктивные члены конъюнкций, антецеденты кванторных импликаций и т.п. До тех пор, пока утверждение используется как сопровождающее по о.д.з., попытки изменить или удалить его будут автоматически блокироваться. Впрочем, приему предоставляется возможность отменять такую блокировку, принимая на себя всю ответственность за последствия. Обычно это делается для преобразований, которые синхронно будут изменять и сопровождаемый, и сопровождающий термы.

Комментарий (сопровождение ...) создается в самом начале решения задачи, сразу после обращения к процедуре "одз". Для этого служат приемы, которые легко найти в пунктах "Ввод комментария СОПРОВОЖДЕНИЕ" разделов оглавления программ, относящихся к общим приемам задач на описание, преобразование, и т.д. Приемы обращаются к процедуре "сопровождтерм", которой передается ссылка на конкретную посылку либо условие P. Просматриваются вхождения в P неоднобуквенных подтермов t, и с помощью справочника "одз" находятся сопровождающие утверждения, отличные от указателей типов объектов. Если множество Q таких утверждений непусто, то предпринимается попытка усмотреть их истинность из контекста вхождения t с помощью проверочных операторов. Эта попытка определяет множество M утверждений контекста, следствием которых служит Q, и далее пара $(M\ t)$ регистрируется в комментарии (сопровождение ...).

Учет информации об исходных посылках, использованных при решении задачи

Прежде чем продолжить описание приемов, остановимся на одном важном элементе структуры данных задачи. Во многих случаях бывает нужно не только получить ответ, но и распознать, какие именно исходные посылки задачи фактически были использованы при его получении. Чтобы отслеживать использование исходных посылок, применяются комментарии (выводимо A), автоматически создаваемые и корректируемые практически всеми приемами решателя. Различаются два типа комментария (выводимо A). Во-первых, комментарий ко всей задаче (кроме задач на исследование). В этом случае A представляет собой список исходных посылок задачи, использованных при преобразованиях ее исходных условий к текущему виду. Во-вторых, комментарий к отдельной посылке f. В этом случае A - список всех исходных посылок задачи, использованных при получении f. Таким образом, отсутствие комментария (выводимо ...) означает, что f - исходная посылка.

Если прием обращается к пакету продукций, то последний выдает не только ответ, но и список посылок, использованных при его получении. Этот список в случае проверочного оператора или синтезатора является значением одной из выходных переменных пакета; в случае нормализатора для накопления данного списка используется комментарий того же вида (выводимо A), что и выше. Отсутствие такого комментария может нарушить работу нормализатора, и его следует вводить даже в тех случаях, когда список A далее не используется.

Для создания и коррекции комментариев (выводимо ...) служит процедура "учетпосылок(x1 x2 x3 x4)". У нее x1 - текущая задача; x2 - список использованных приемом утверждений текущего контекста. Значение x4 либо равно 0, и тогда значением x3 является вхождение выведенной либо преобразованной приемом посылки, либо равно 1, и тогда x3 игнорируется. Выполняется создание либо коррекция комментария (выводимо ...): в первом случае - к посылке x3, во втором - к задаче x1. Процедура отбирает те элементы списка x2, которые являются посылками задачи x1. Затем, с помощью комментариев (выводимо ...) к этим посылкам, определяется множество M исходных посылок, на которые опирается текущий прием. После этого остается лишь пополнить старый комментарий (выводимо ...) элементами списка M либо создать новый такой комментарий. Обращаться к процедуре "учетпосылок" следует до того, как прием фактически изменил посылку x3. Заметим, что при использовании ГЕНОЛОГа обращения к этой процедуре создаются компилятором автоматически, без ввода в описание приема каких-либо специальных указателей.

Противоположные посылки

Если задача на описание имеет две посылки, одна из которых является отрицанием другой, то ее список посылок противоречив, и можно считать ответом на задачу логическую константу "истина", представленную в формате терма. Прием, который отслеживает указанную ситуацию, активизируется на нулевом уровне сканирования. Он просматривает все посылки, имеющие вес 0 (т.е. только что измененные либо занесенные посылки), и для каждой выясняет, содержится ли ее отрицание в списке посылок. Так как обоснованием ответа "истина" служат только найденные две противоположные посылки, то прием сбрасывает ранее имевшуюся в комментарии (выводимо ...) информацию об исходных посылках, использованных при решении, и переустанавливает ее по указанным двум посылкам.

Идентичные посылки

Если задача на описание имеет две идентичные посылки, то одна из них отбрасывается, а все комментарии отброшенной посылки добавляются к комментариям оставшейся. Прием применяется на уровне 0; он просматривает все посылки задачи, имеющие вес 0, и проверяет наличие идентичной посылки только для них. Это приводит к тому, что обычно отбрасывается та посылка, которая появилась позднее.

Условие, имеющееся в посылках

Если условие задачи, отличное от константы "истина", содержится в посылках, то оно отбрасывается, а в комментарии (выводимо ...) регистрируется равная ему посылка. Исключение составляют только дизъюнктивные условия, сопровождаемые комментарием "разборслучаев", т.е. введенные специально для рассмотрения подслучаев. Заметим, что невозможно удалить вообще все условия задачи на описание:

когда остается только одно условие, то вместо удаления предпринимается замена его на константу "истина".

Идентичные условия

При усмотрении совпадающих условий задачи на описание одно из них отбрасывается вместе со своими комментариями. Схема поиска совпадения та же, что для посылок - просматриваются все условия F веса 0, и если имеется другое условие, равное F, то удаляется F. Уровень срабатывания приема нулевой.

Посылка является частью условия

Если посылка A имеет вхождение в условие B, причем это вхождение не расположено в области действия кванторов либо описателей по свободным переменным утверждения A, то данное вхождение заменяется в условии B на константу "истина". Прием просматривает все посылки веса 0, и для каждой такой посылки - все условия. Уровень его срабатывания равен 0.

Для изменения условия прием использует процедуру "заменавхождения". Эта процедура относится к числу немногих процедур, применяемых приемами для преобразования задачи. Коротко о ней рассказывалось в первой книге, однако для развития решателя необходимо более детальное представление о ее устройстве. Мы рассмотрим эту процедуру в следующей главе, наряду с рядом других общих процедур, используемых приемами. Локализация всего многообразия действий решателя по изменению контекста внутри немногих процедур позволяет вводить дополнительные механизмы управления и сопровождения. В отличие от логики принятия решений, рассредоточенной по конкретным приемам, эти механизмы позволяют отследить или заблокировать действия решателя из каких-либо совсем общих соображений, а также автоматически дополнить основные действия приема рядом сопутствующих действий. Они существенно усложняют программы "преобразующих" процедур, составляя главную часть их объема и трудоемкости.

Упрощение посылок

Если задача имеет невырожденные посылки, которые до этого не анализировались, то имеет смысл в начале решения попытаться упростить эти посылки, привести их к стандартному для решателя виду. Таким образом может сделаться явной та или иная замаскированная вначале информация.

Прием, предпринимающий упрощение посылок задачи на описание, активизируется при текущем уровне 2 либо 3. Сначала он рассматривает целевую установку. Если из нее ясно, что посылки уже были ранее упрощены, или что их анализ целесообразно отложить до обращения к подзадаче, то дальнейшие действия блокируются. Например, не упрощаются посылки задачи, имеющей цель "редакция", так как в ней происходит обработка найденного ответа, а посылки должны были анализироваться до этого. Затем просматриваются все посылки A веса 2. Если задача не имеет цели "прямойответ", то из просмотра исключаются посылки с квантором существования в заголовке, так как они будут преобразованы путем ввода вспомогательных объектов. Не рассматриваются также элементарные посылки, имеющие лишь однобуквенные подвыражения. Попытка упрощения посылки выполняется однократно - уже рассмотренные посылки снабжаются комментарием "упростить". Для упрощения используется обращение к вспомогательной задаче на преобразование. Условием этой задачи служит текущая посылка A, а посылками - все оставшиеся посылки задачи на описание. Если результат упрощения B отличен от A, то процедура "заменавхождения" заменяет посылку A на B.

Анализ объединенного списка условий и посылок

Ранее уже говорилось, что многие задачи на описание решаются с помощью вывода следствий из объединенного списка посылок и условий. Этот объединенный список хранится в задаче на исследование A, связываемой с задачей на описание Z и называемой ее блоком анализа. Он представляет собой список посылок задачи A. Обычно новая задача на описание не имеет блока анализа. Та позиция ее структуры данных, которая хранит ссылку на блок анализа (последний разряд набора), вначале заполняется либо пустым словом, либо, если блок анализа нежелательно создавать вообще, нулем. В процессе рассмотрения задачи решатель создает блок анализа самостоятельно. Он может делать несколько попыток развития данного блока - с возрастающей последовательностью уровней обращения, привлекая каждый следующий раз все более и более мощные средства.

Для первоначального ввода блока анализа и организации обращений к нему служит один и тот же прием. Если задача на описание Z имеет цель "исследовать", означающую, что ответом на нее являются все представляющие интерес утверждения, выведенные в блоке анализа, то уровень активизации приема равен 1. В противном случае он активизируется не ранее уровня 4.

Прием создает вспомогательную задачу на исследование A, регистрируя ее как блок анализа задачи Z, либо берет уже созданный ранее блок анализа A. Он обращается к решению задачи A, определяя максимальный уровень обращения в зависимости от контекста. От контекста же зависит уровень срабатывания самого приема. Если при рассмотрении блока анализа оказываются найдены следствия, требующие переосмысления на уровне внешней задачи Z, то они передаются в список условий последней, и работа с задачей A обрывается. Обрыв решения задачи A и возвращение к задаче Z инициируется оператором ЛОСа "ответ(A)".

Опишем действия приема несколько более подробно. После проверки того, что последний разряд набора, представляющего текущую задачу на описание Z, отличен от 0, а цели и комментарии этой задачи не отменяют рассмотрения блока анализа, составляется список всех логических символов, входящих в условия задачи Z. По нему определяется список S разделов, к которым относятся данные символы. Здесь используется процедура "разделы", обращающаяся к справочнику "раздел". Последний справочник, а также справочник "содержание" регулярно пополняются по мере ввода новых понятий при обучении решателя.

Знание разделов, к которым относится задача Z, позволяет определить как величину текущего уровня сканирования, на котором следует обращаться к блоку анализа, так и максимальный уровень обращения к нему. Для этого служит справочник "вход". Ему сообщаются название раздела, задача Z и текущий уровень сканирования. Если прием справочника находит целесообразным рассмотрение блока анализа, то выдает ненулевую величину максимального уровня обращения. Находится максимум M таких величин (в программе приема - значение x7). При наличии у Z целей "известно" либо "исследовать", которые непосредственно указывают на использование блока анализа как главного средства решения задачи, значение M сразу полагается наибольшим - равным 16.

Если блок анализа A еще не создан, то он создается как задача на исследование, имеющая своим списком посылок объединение списков условий и посылок задачи Z. Веса всех посылок полагаются равными 0; условия f задачи Z сопровождаются в этом списке комментариями (условие f (f)). Комментарий (условие g K) к посылке f впоследствии будет отслеживать возможные изменения этой посылки, указывая, что условие g задачи Z является следствием f и прочих посылок задачи A, перечисляемых наряду с f в списке K. В качестве общих комментариев к списку посылок задачи A вводятся: (описать B), где список B перечисляет все условия задачи Z, перенесенные в A; (вход C), где C - обратная ссылка из A на задачу Z, а также "обращение". Заметим, что в качестве ссылки C здесь берется не сама задача Z, а вхождение первого разряда представляющего ее набора. Это - мера предосторожности против возникновения циклических ссылок: ведь из Z имеется ссылка непосредственно на A. Циклические ссылки в зоне задач недопустимы, так как они могут приводить к зависанию процедуры ЛОСа, сравнивающей два набора. Комментарий "обращение" инициирует выдачу на экран в режиме просмотра решения информации об обращении к блоку анализа A. В список целей задачи A переносятся из списка целей задачи Z набор (неизвестные ...), перечисляющий неизвестные, а также ряд других целей, уточняющих режим вывода следствий.

Если задача Z имела цель (известно $a_1 \dots a_n$), перечисляющую переменные $a_1, \dots, a_n$, через которые должен быть выражен ответ, то она не переносится в задачу A; вместо нее задаче A передается цель "известно". В этой ситуации все переменные посылок задачи Z, не упомянутые среди $a_1, \dots, a_n$, становятся дополнительными неизвестными задачи A. С другой стороны, неизвестные задачи Z, которые уже оказались неявно выражены ее условиями через $a_1, \dots, a_n$, исключаются из неизвестных задачи A.

Если задача на описание Z представляет собой подслучай, возникший ранее при рассмотрении некоторой дизъюнктивной посылки, то задача A дополняется целью "контроль". Эта цель блокирует поспешную выдачу ответа, найденного в A для рассматриваемого подслучая, так как сам подслучай может оказаться невозможным. Лишь по достижении уровня 3, когда попытки усмотрения противоречивости списка посылок задачи A окажутся безуспешными, ответ будет выдан.

Задаче A передается ряд специальных комментариев задачи Z, например, комментарий (геомредактор ...), определяющий текущий чертеж.

По завершении пополнения структуры данных задачи A, она регистрируется в качестве блока анализа задачи Z, и предпринимается обращение к ее решению до максимального уровня M. Окончание рассмотрения задачи A возможно по двум причинам - либо по исчерпании допустимого уровня, либо при обрыве, вызванном действиями какого-либо приема, обратившегося к оператору ЛОСа "ответ(A)". В последнем случае тот же прием модифицирует и задачу Z, так что она уже содержит в себе найденный при рассмотрении A результат. Поэтому далее возобновляется сканирование задачи Z; если она была изменена, то текущий уровень сканирования предварительно понижается до 0. Чтобы предотвратить повторные попытки обращения к блоку анализа с одним и тем же максимальным уровнем M, используются комментарии (вход M) к задаче Z.

Если на момент принятия решения о рассмотрения блока анализа A этот блок уже был создан, то предпринимается пополнение списка посылок задачи A теми условиями задачи Z которые ранее не были перенесены в A. Далее, как и выше, происходит обращение к задаче A с максимальным уровнем M.

Разбиение условий на независимые группы

Если множество содержащих неизвестные условий задачи на описание Z разбивается на такие подгруппы $M_1, \ldots, M_n$, что утверждения из разных подгрупп не имеют общей неизвестной, то можно перейти к решению независимых задач $Z_1, \ldots, Z_n$ со списками условий $M_1, \ldots, M_n$. К каждому из этих списков добавляются также все условия задачи Z, вообще не содержащие неизвестных. Ответ на задачу Z будет представлять собой объединение ответов на указанные задачи. Прием, выполняющий данные действия, нетрудно найти по оглавлению программ. Его программа начинается с контрольной точки "прием(13)" в программе символа "описать". Уровень срабатывания приема равен 2.

Во-первых, предпринимается проверка целесообразности разбиения списка условий на независимые группы. Например, оно не выполняется для задач с целью "редакция" или "известно ...", а также для задач, имеющих комментарий (контекст ...). Последний комментарий возникает при исключении части неизвестных задачи, для которых было найдено явное их описание через другие неизвестные. При редактировании ответа в каком-либо отдельном подслучае из комментария извлекаются утверждения, связывающие ранее исключенные неизвестные с оставшимися. Эти утверждения присоединяются к прочим условиям задачи, и упрощение относится ко всему списку утверждений, определяющих подслучай. Таким образом, при последующей склейке подслучаев в общий ответ не нужно специально учитывать ранее исключенные неизвестные. Разбиение на компоненты связности нарушило бы данный режим, из-за чего оно и блокируется.

Далее определяется разбиение списка содержащих неизвестные условий задачи Z на компоненты связности M_i по зависимости от общей неизвестной. Находится также список M_0 всех условий, не содержащих неизвестных. Составляется список S пар ($M_i \cup M_0$ - множество неизвестных, входящих в утверждения из M_i). В программе приема этот список присвоен переменной x6. Если число компонент более одной, то они последовательно просматриваются, и для каждой из них создается задача Z_i. Комментарии к условиям задачи Z_i, а также комментарии ко всей этой задаче по умолчанию берутся из Z. При необходимости они могут быть скорректированы с помощью справочника "копияфайла". Это же относится и к целям задачи, где используется другой справочник - "сокращнеизв". Для отображения (при трассировке) на экране факта обращения к задаче Z_i, она снабжается комментарием "обращение". Предпринимается обращение к решению задачи Z_i, причем максимальный уровень - тот же, что у задачи Z. Если получен отказ, то и на задачу Z выдается отказ. Иначе - список конъюнктивных членов ответа задачи Z_i и список использованных при получении этого ответа исходных посылок задачи Z регистрируются в специальном накопителе (R_1, R_2). В программе приема роль этого накопителя играет переменная x7.

После решения всех задач Z_i проверяется наличие цели "упростить" у задачи Z. Если она есть, то создается вспомогательная задача на описание Z', условия которой суть все элементы накопителя R_1, а цели - дополненные символом "редакция" цели задачи Z. Эта задача решается; в комментарии (выводимо ...) задачи Z регистрируются все исходные посылки, использованные в процессе решения $Z_1, \ldots, Z_n, Z'$, и выдается ответ задачи Z'. Если же цели "упростить" у задачи Z не было, то в качестве ответа выдается конъюнкция утверждений списка R_1.

Непосредственный подбор значений неизвестных

Если в задаче на описание достаточно получить лишь пример значений неизвестных, удовлетворяющих ее условиям, то можно проверить, не содержат ли посылки группу утверждений, непосредственно дающих такой пример, т.е. получающихся из условий подстановкой вместо неизвестных каких-то конкретных выражений. При этом следует учитывать, что искомые посылки могут идентифицироваться с условиями лишь после применения к ним каких-либо простых преобразований, например, перестановки операндов в коммутативных операциях и отношениях. Иногда достаточно лишь частичной идентификации условий с посылками, позволяющей однозначно определить предполагаемые значения неизвестных, после чего легко усматривается истинность оставшихся условий.

Такого рода сравнение условий с посылками нужно проводить быстро, не доводя дело до сколь-нибудь значительного перебора, так как случаи, в которых оно вообще оказывается результативным, достаточно редки. С другой стороны, задачи, где ответ все же усматривается указанным способом, относятся к категории "очевидных", и данный прием должен применяться на малых уровнях.

В решателе имеется несколько приемов, выполняющих попытку непосредственного сопоставления условий с посылками для усмотрения ответа. Прежде всего, эта попытка предпринимается на уровнях 1 и 3. Проверяется, что все условия задачи суть элементарные утверждения, и для подбора значений неизвестных используется процедура "подборзначений". Она находит условие, заголовок которого (с учетом внешнего отрицания) встречается в посылках наименьшее число раз. Последовательно рассматриваются варианты идентификации этого условия с посылками; для идентификации оставшихся условий предпринимаются рекурсивные обращения к той же самой процедуре. Если удалось подобрать значения неизвестных, то перед выдачей ответа проверяется наличие комментария (контекст ...). Он хранит ту часть ответа, которая связана с ранее исключенными (выраженными через оставшиеся) неизвестными. Эта часть присоединяется к текущему ответу, причем в нее подставляются значения найденных неизвестных.

На уровне 4 предпринимается еще одна попытка подбора значений неизвестных. В ней учитывается возможность наличия условий без неизвестных. Для таких условий выполняются попытки усмотрения их истинности с помощью вспомогательных задач на доказательство, решаемых с максимальным уровнем 4. В случае успеха используется процедура "подборнеизвестных", идентифицирующая с посылками остальные условия. Чтобы все эти действия выполнялись быстро, введен ограничитель трудоемкости, по исчерпании которого применение приема обрываются.

Еще одна попытка подбора, по существу совпадающая с предпринимавшейся на уровнях 1 и 3, реализуется на уровне 5.

Наконец, упомянем еще один прием, несколько отличающийся по своему назначению от указанных выше. Если задача на описание имеет цели "пример", "полный", причем все ее неизвестные - несущественные, т.е. содержатся в цели (параметры ...), то она фактически эквивалентна задаче на доказательство существования. Предположим, что удалось подобрать такие значения ее неизвестных, для которых все условия, кроме одного, отобразились в некоторые посылки. Обозначим A результат подстановки в оставшееся условие найденных значений неизвестных; пусть этот результат не содержит неизвестных. Тогда можно решать задачу путем разбора случаев, присоединяя к ее посылкам утверждение $A \vee \neg A$. Первый подслучай очевиден - необходимые значения неизвестных существуют. Поэтому остается только рассмотрение второго подслучая. Однако, технически проще избежать явного рассмотрения случаев, сразу же добавляя к списку посылок задачи утверждение $\neg A$. Конечно, это повлияет на итоговые значения неизвестных - они должны были бы определяться условными выражениями, в зависимости от истинности A. Но так как неизвестные несущественны, данное обстоятельство не играет никакой роли. Прием, выполняющий описанный "вывод" новой посылки $\neg A$, активизируется на уровне 4. Для подбора неизвестных он использует процедуру "частичнподбор".

Выражение одной неизвестной через другие

Если задача на описание имеет более одной неизвестной, то можно сначала попытаться, выделив какую-то одну неизвестную x и временно считая все остальные неизвестные известными, начать решение относительно x. Получив ответ и исключив из него ту часть, которая определяет допустимые значения x, далее можно вернуться к решению задачи относительно оставшихся неизвестных. По завершении последнего процесса остается объединить ответы и упростить результат. Эта схема часто применяется в математике; классический пример - решение систем линейных уравнений методом Гаусса. Однако, даже в элементарной алгебре она не является универсальной. Легко привести примеры, в которых после выражения одной неизвестной через другие система уравнений становится чрезмерно громоздкой, и задача заводится в тупик. При обучении решателя прием, реализующий данную схему, накопил в себе настолько существенные ограничения, что в некоторых разделах оказался почти отключенным. Например, в элементарной алгебре он применяется лишь для случаев, когда число уравнений системы меньше числа неизвестных. Разумеется, если система имеет одно или несколько линейных уравнений, то решатель попытается использовать их для исключения неизвестных. Однако, эти действия выполняются не общим приемом, а специальными реализованными на ГЕНОЛОГе приемами, относящимися именно к линейным уравнениям. Даже они имеют множество дополнительных ограничений. После того, как в задаче появляется уравнение вида $x = t$, явно выражающее неизвестную x через прочие неизвестные, неизвестная x исключается приемом, закрепленным за символом "равно" (см. ниже).

Впрочем, указанный выше прием неплохо работает в других разделах; например, при решении систем уравнений для множеств.

Перейдем к более подробному описанию действий приема. Программу его можно найти в пункте "Решение задачи на описание с несколькими неизвестными путем выражения одной из неизвестных через остальные" оглавления программ.

Сначала предпринимается анализ целевой установки приема. Проверяется отсутствие цели "редакция", означающей, что задача уже решена и выполняется редактирование ответа; отсутствие цели "пример"; наличие цели "полный"; отсутствие комментария "блокнеизвестных", блокирующего повторную попытку применения приема; отсутствие дизъюнктивных условий, для которых будут разбираться подслучаи, и т.п.

Если из целевой установки и общего вида условий не усматривается нецелесообразность применения приема, то находится список S всех разделов, к которым относятся встречающиеся в задаче логические символы. Просматриваются входящие в S символы f, и для окончательного принятия решения о применении приема выполняются обращения к справочнику "описать" на этих символах. Здесь же уточняется уровень применения приема - 2 либо 3. Если попытка признается целесообразной, то справочник выдает список всех неизвестных x, для которых ее следует выполнять.

Неизвестные в этом списке упорядочены по убыванию приоритета. Берется первая неизвестная x, и для нее создается вспомогательная задача Z'. Она получена из текущей задачи Z преобразованием целей, соответствующим сохранению единственной неизвестной x. Такое преобразование осуществляется с помощью справочника "сокращнеизв". Задача Z' снабжается комментарием (сокращнеизв Z^* X), где Z^* - исходная задача со многими неизвестными, совпадающая с Z либо (если уже выполнялось исключение неизвестных) извлекаемая из комментария (контекст $Z^* \ldots$); X - список неизвестных, временно рассматриваемых как известные. Хотя вспомогательная задача Z' изначально ориентирована на разрешение условий только относительно x, в процессе ее решения комментарий (сокращнеизв $\ldots$) будет учтен таким образом, чтобы получить сразу ответ на задачу Z. Как именно это происходит, опишем ниже. Здесь же заметим, что прием обращается к решению задачи Z'. Если на нее получается ответ, отличный от символа "отказ", то он выдается как ответ на Z. Иначе - вводится комментарий "блокнеизвестных" и выполняется откат к просмотру следующего символа f.

В процессе решения задачи Z' может происходить разбор случаев, и комментарий (сокращнеизв $\ldots$) будет передаваться задачам, соответствующим отдельным подслучаям. После получения окончательного ответа для x в некоторой такой задаче Z'', решатель обращается к процедуре "редакторответа", выполняющей завершающую обработку ответа. Здесь и реализуется учет комментария (сокращнеизв Z^* X). Программу, выполняющую этот учет, можно найти в пункте "Приемы решателя" - "Общие процедуры, используемые в приемах" - "Процедура РЕДАКТОРОТВЕТА" - "Учет неизвестных, которые временно рассматривались как известные" оглавления программ. Ответ представляется в виде $A(x)$ & B, где B - все утверждения без неизвестной x. Создается вспомогательная задача Z''' для разрешения относительно оставшихся неизвестных X условий, полученных добавлением к B условия существования $\exists_x A(x)$, если оно не очевидно. В программе эта задача присваивается переменной x10. Информация о фрагменте ответа $A(x)$ передается задаче Z'' через комментарий (контекст $\ldots$), который может также содержать информацию о других ранее исключенных неизвестных. После этого предпринимается решение задачи Z'''. Если на нее получен "отказ", то выдается отказ на Z''. Так как при редактировании ответов отдельных подслучаев, возникающих в Z''', будет выполняться присоединение и упрощение фрагментов ответа, сохраненых в комментарии (контекст $\ldots$), то получение ответа R на Z''' одновременно будет означать получение ответа и на Z (разумеется, лишь для текущего подслучая, определяемого задачей Z''). В этой ситуации будет предпринято перенесение тех комментариев задачи Z''', которые представляют ценность для внешних задач, в задачу Z'', и выдан ответ R задачи Z''. Обработка комментариев выполняется здесь справочником "преобразование".

Попытка исключения несущественной неизвестной

У задачи на описание, имеющей цель (параметры $x_1 \ldots x_n$), неизвестные $x_1, \ldots x_n$ являются несущественными - они не обязаны входить в ответ. Если усматривается, что существование значения какой-либо несущественной неизвестной x, для которого истинны все содержащие ее условия $A_1, \ldots, A_n$, является следствием прочих условий, то данная неизвестная и все условия на нее могут быть из задачи исключены. Несущественную неизвестную x можно исключить и другим путем - если усмотреть, что утверждение $\exists_x(A_1$ & $\ldots$ & $A_n)$ эквивалентно какому-то простому бескванторному утверждению B и заменить $A_1, \ldots, A_n$ на B. В простейших случаях такое усмотрение выполняется приемами, ориентированными на конкретные ситуации. Например, если все содержащие несущественную неизвестную x условия суть $a < x$, $x < b$, "x-число", то они заменяются на $a < b$.

Если задача не требует получения полного ответа - либо имеет цель "пример", либо не имеет цели "полный", то предусмотрен общий прием исключения несущественной неизвестной (см. пункт "Попытка исключения несущественной неизвестной путем решения вспомогательной задачи на описание с единственной неизвестной"). Он проверяет, что число неизвестных задачи не менее 2, и выбирает какую-либо несущественную неизвестную x. Для блокировки повторных рассмотрений проверяется отсутствие комментария (свертка x), который затем сразу же вводится. Создается вспомогательная задача Z', полученная из текущей задачи Z перенесением в посылки всех не содержащих x условий и выбором x в качестве единственной неизвестной, причем несущественной. Она решается с умеренным ограничением на допустимую трудоемкость. Если получен ответ B, отличный от символа "отказ", то проверяется отсутствие x в этом ответе и происходит регистрация всех его конъюнктивных членов в условиях задачи Z вместо всех старых условий, содержавших x. Переменная x исключается из неизвестных, и веса условий задачи Z полагаются равными 0.

Упрощение выражений при редактировании ответа

После того, как усмотрен ответ задачи на описание, обычно применяется процедура "редакторответа", выполняющая упрощение ответа в соответствии с целевой установкой задачи. Подробнее эта процедура будет рассмотрена позднее. Пока же отметим, что для упрощения ответа используется вспомогательная задача на описание, имеющая цель "редакция". В ней ответ определяется уже без обращения к процедуре "редакторответа", по исчерпании отведенных для решения средств.

При решении задачи, имеющей цель "редакция", предпринимается попытка упростить подвыражения, не содержащие неизвестных. Это важно, так как после перехода к утверждениям, образующим требуемое описание значений неизвестных, решение задачи сразу прекращается, и относительно известных параметров указанные утверждения могут оказаться совершенно необработанными. Упрощение подвыражения, содержащего только известные параметры, выполняется в два этапа: сначала (на текущем уровне 2) с помощью вспомогательных задач на преобразование; затем (на уровне 3) с помощью процедуры "свертка", обеспечивающей переход к возможно более короткой записи. Эта процедура обращается к специальным пакетным нормализаторам, заголовки которых получаются добавлением префикса "упрощ" к названию корневой операции. Они легко находятся по оглавлению базы приемов. В качестве примера приведем прием пакета "упрощумножение", преобразующий произведение $a^c b^c$ к виду $(ab)^c$. Заметим, что этот переход нужен лишь для сжатия ответа; в процессе решения задачи обычно применяется обратное преобразование, позволяющее получать одночлены стандартного вида.

К программе приема, выполняющего упрощение известных подвыражений при редактировании ответа, можно перейти через оглавление программ. Уровни срабатывания приема - 2 и 3. После анализа целевой установки задачи (в частности, проверки наличия цели "редакция") принимается решение о применении приема. Просматриваются условия F, не используемые для сопровождения по о.д.з. и не имеющие комментария (параметры U), указывающего, что на текущем уровне U известные подвыражения F уже упрощались. Проверяется отсутствие более короткого условия, подходящего для упрощения известных подвыражений и еще не помеченного данным комментарием. Указанный комментарий к F вводится. Составляется список S максимальных известных подвыражений утверждения F. Так как в ответе задачи с несколькими неизвестными встречаются утверждения, выражающие одну неизвестную через другие, локально как бы известные, то при составлении списка S это учитывается, и в него могут попасть выражения с "побочными" для F неизвестными. Если встречается ассоциативно-коммутативная операция, то в S отбирается ее фрагмент, образованный всеми известными операндами. Из S исключаются однобуквенные выражения и десятичные числа. Для уровня 2 отбрасываются также подвыражения ранее упрощенных выражений A. Последние распознаются по комментариям (упрощение A). В программе приема список S присвоен переменной x9. Далее выполняется просмотр элементов t списка S. Если текущий уровень равен 2, то организуется обращение к вспомогательной задаче на упрощение t относительно контекста утверждения F. Для результата T создается комментарий (упрощение T). Если T отличается от t, то выполняется замена t на T. Она относится не только к F, но сразу ко всем содержащим t термам задачи. После выполнения замены просмотр списка S продолжается. Действия в случае уровня 3 аналогичны, но вместо обращения к вспомогательной задаче предпринимается обращение к процедуре "свертка". По окончании просмотра списка S - откат к началу цикла сканирования задачи с обнулением текущего уровня (оператор "пересмотр").

Выдача ответа

В большинстве случаев ответ задачи на описание выдается приемами ГЕНОЛОГа, усматривающими в списке условий явное описание для единственной неизвестной. Перечислим те особые случаи, для которых предусмотрен общий прием выдачи ответа, реализованный на ЛОСе. Перейти к их программам можно через раздел "Приемы решателя" - "Общие приемы" - "Задачи на описание" - "Выдача ответа" оглавления программ.

1. Если задача имеет цель "редакция" и достигнут ее максимальный уровень, то проверяется, не противоречит ли целям задачи вид ее списка условий. Для этого используется справочник "редакция", обращения к которому происходят на символах - заголовках целей. Если противоречия не усмотрено, в качестве ответа выдается конъюнкция условий задачи.

2. Если задача имеет единственное условие $P(x)$, где x - неизвестная, а P - название одного из основных типов объектов ("число", "множество", "функция", и т.п.), то на текущем уровне 0 выдается ответ. Если задача не имеет цели "редакция", то выдача ответа осуществляется процедурой "редакторответа", иначе - ответ $P(x)$ выдается непосредственно. При наличии цели "пример" прием блокируется, так как в этом случае нужно указать конкретное значение x. Обращение к процедуре "редакторответа" при отсутствии цели "редакция" необходимо не для упрощения и без того "минимального" ответа, а для учета информации о внешем контексте, которая, возможно, потребует присоединения к ответу утверждений, характеризующих ранее исключенные неизвестные.

3. Если задача имеет цель "стоп", то ответ на нее выдается на текущем уровне 0 и представляет собой конъюнкцию условий. Такая ситуация складывается, если задача нужна только для получения списка утверждений, пополненного ограничениями на о.д.з.

4. Если все условия задачи оказались не содержащими неизвестных, то в качестве ответа выдается конъюнкция этих условий, возможно, предварительно упрощенная. Уровень срабатывания приема равен 0. Проверяется, что задача не имеет целей "редакция" и "независит ..." (последний случай обрабатывается другим приемом). Отбрасывается случай задачи без неизвестных, имеющей цель "явное", если только ее список условий не состоит из единственной логической константы "истина" или "ложь". Если задача имеет цели "полный", "пример", то предпринимается попытка усмотреть истинность всех ее условий, и при неудаче выдача ответа блокируется. В прочих случаях, в зависимости от наличия цели "упростить", либо происходит обращение к процедуре "редакторответа", либо сразу выдается конъюнкция условий. Особо анализируется случай наличия в условиях константы "ложь" - тогда она и выдается в качестве ответа.

5. Если задача имеет одну из целей "мощность", "исследовать", "попыткаспуска", то ответ на нее выдается по достижении максимального уровня. Для этого используется процедура "редакторответа".

6. Если задача имеет цель "свобоперанд", то ответ на нее выдается сразу же, как только в ее условиях пропадают кванторы. Прием, отслеживающий это событие, применяется на уровне 0. Ответом служит конъюнкция условий.

7. Если задача имеет комментарий "ответ", то сразу же (на уровне 0) в качестве ответа выдается конъюнкция условий. Комментарий обычно используется для передачи уже найденного ответа из вспомогательной задачи во внешнюю задачу. Для этого вспомогательная задача должна изменить список условий внешней задачи и передать ей указанный комментарий.

8. Если задача имеет цель "перечисление", то она решается в режиме перечисления частичных ответов, накапливаемых в наборе A комментария (ответзадачи A). Регистрация этих ответов выполняется процедурами "попыткаспуска", "попыткапараметризации", используемых приемами при попытках решить задачу в частных случаях. По достижении максимального уровня составляется дизъюнкция утверждений накопителя A, на которую заменяются условия задачи и которая выдается в качестве ответа. При пустом списке A выдается отказ.

9. Для задачи, имеющей только цели "независит" и "прямойответ", ответ выдается по достижении максимального уровня, если ее условия не содержат запрещенных переменных. Аналогичные действия выполняются для задачи, имеющей цели "длялюбого" и "независит", однако уже на нулевом текущем уровне.

10. Предусмотрен специальный прием для выдачи ответа на дифференциальные уравнения, если производные неизвестных функций исключены, но явно разрешить полученные соотношения относительно этих функций не удалось.

11. Совсем особый случай - задачи с целью "вычисление". Такие задачи предназначены для создания программы ЛОСа, находящей значения неизвестных по заданным значениям известных параметров. Сначала задача решается как обычная задача на описание, причем в списке ее условий складывается схема вычислений, понятная компилятору ГЕНОЛОГа. Она представляет собой

некоторую совокупность утверждений, последовательно определяющих новые значения через ранее определенные. В этом отношении она ничем не отличается от списка антецедентов обрабатываемой компилятором ГЕНОЛОГа теоремы приема. Поэтому, по завершении решения "обычной" задачи на описание, остается лишь обратиться к компилятору для получения исходной программы. Подробнее все эти действия будут рассматриваться в главе, посвященной вычислительным задачам. Пока ограничимся указанием на пункт "Компиляция ответа задачи на вычисление" рассматриваемого подраздела оглавления программ, через который можно выйти на начало программы приема, обращающегося к компилятору ГЕНОЛОГа по завершении подготовки схемы вычислений. Прием активизируется на максимальном уровне, т.е. по исчерпании средств, отведенных для обработки указанной схемы. Собственно обращение к компилятору выполняется процедурой "вычисление", которая регистрирует созданную программу в блоке программ как программу справочника "вычисление", закрепленную за некоторым (выбранным наугад) ключевым логическим символом s. Если задача не имела известных параметров, то вычисления выполняются немедленно, путем обращения к справочнику "вычисление" на символе s. По итогам формируется список равенств, указывающих найденные значения неизвестных. Условия задачи заменяются на эти равенства, и конъюнкция их выдается как ответ. Если же задача имела известные параметры, то ответом на нее служит однобуквенный терм "программа". Однако, перед выдачей данного ответа на экран произойдет обращение к интерфейсу, предоставляющему возможность вводить нужные значения параметров и вычислять для них значения неизвестных. Ответ будет прорисован лишь после выхода из этого интерфейса. Заметим, что в обоих случаях созданная программа справочника "вычисление" сохраняется, и дальнейшие обращения к ней можно выполнять без повторного запуска решения задачи.

Задача на проверку истинности

Задача на установление истинности либо ложности утверждения оформляется как задача на описание, имеющая своим условием это утверждение и не имеющая неизвестных. Такая задача сопровождается целью "проверка". Сначала она решается в режиме эквивалентных преобразований условия - в простейших случаях это уже может привести к получению константы "истина" либо "ложь". Однако, по достижении уровней 7 и 10 предпринимаются явные попытки доказать либо опровергнуть утверждение с помощью вспомогательной задачи на доказательство. На уровне 7 при обращении к вспомогательной задаче устанавливается лимит трудоемкости, в несколько раз меньший лимита, устанавливаемого на уровне 10.

Задача с целью "замещение"

В некоторых ситуациях бывает необходимо найти переформулировку группы утверждений относительно заданного контекста, при который были бы использованы только переменные заданного списка. Такие ситауции возникают, например, при решении задач на поиск экстремальных значений параметров, допускающих некоторый "сценарий" - геометрический чертеж, физический процесс и т.п. Здесь необходимо сначала перевести качественные условия, описывающие допустимость сценария, на язык соотношений для числовых параметров, а затем уже применять традиционные методы анализа.

Задача на описание, в которой нужно преобразовать условия к заданному списку переменных, снабжается целью "замещение". Неизвестными ее служат все переменные условий, не относящиеся к этому списку. Чтобы было возможно задать связи неизвестных с известными, допускается вхождение первых не только в условия, но и в посылки задачи.

Прием, обращающийся к задаче с целью "замещение" для отыскания экстремальных значений, будет рассмотрен в другом разделе. Здесь же рассмотрим лишь прием, который предпринимает попытку выразить атомарные числовые подвыражения условий, содержащие неизвестные, через известные числовые параметры. К программе его можно перейти из пункта "Попытка вычисления неизвестных подвыражений в задаче с целью "замещение" " оглавления программ.

Сначала проверяется отсутствие комментария "замещение", блокирующего повторную попытку применения приема, и этот комментарий сразу же вводится. Затем просматриваются условия F, причем сначала - все равенства, а затем уже остальные условия. Находится список S атомарных числовых параметров условия F, содержащих неизвестные и отличных от переменных. В программе этот список присвоен переменной x10. Если список непуст и состоит из выражений $t_1, \ldots, t_n$, то выбираются новые переменные $x_1, \ldots, x_n$. Находится список $p_1, \ldots, p_n$ всех известных числовых переменных, встречающихся в контексте условия F. Решается вспомогательная задача на описание Z', условия которой суть равенства $x_1 = t_1, \ldots, x_n = t_n$ и указатели типа значения "число(x_1)", ... , "число(x_n)". Посылками ее служат все посылки текущей задачи Z, к которым присоединены не содержащие выражений $t_1, \ldots, t_n$ остальные условия задачи Z. Задача Z' имеет цели (неизвестные $x_1 \ldots x_n$), (известно $p_1 \ldots p_n$), "полный", "явное", "прямойответ". Если найден ответ R задачи Z', определяющий выражения $r_1, \ldots, r_n$ для $x_1, \ldots, x_n$, то во всех условиях задачи Z предпринимается замена выражений t_i на r_i. Далее - продолжение просмотра условий F.

Переход к новой несущественной неизвестной

Если задача на описание имеет такую несущественную неизвестную x, которая встречается (за исключением условия $P(x)$, определяющего тип значения x) только внутри заданного терма t вида $f(\ldots\ x \ldots)$, и этот терм имеет не менее двух вхождений в условия, то имеет смысл сам терм t обозначить новой вспомогательной несущественной неизвестной, а старую неизвестную x не рассматривать вообще. Этот прием реализован для частного случая, когда x и t принимают числовые значения. Его программу можно найти в пункте "Переход к новой несущественной неизвестной" оглавления программ. Несущественная неизвестная x обозначена в программе посредством x7, а вхождение терма t - x12. На роль вспомогательной несущественной неизвестной сначала выбирается новая переменная y (см. переменную x15), и решается задача на описание Z', имеющая условия "число(y)", $\exists_x(y = t)$. Посылками ее служат все посылки текущей задачи Z, а также все ее условия, не содержащие x. Задача Z' имеет неизвестную y и дополнительную цель "см", блокирующую попытки повторного применения данного приема при ее решении (что могло бы привести к зацикливанию). Если на задачу Z' получен ответ R, не содержащий переменной x, то выполняется преобразование задачи Z. При этом преобразовании оказывается удобным сохранить для новой несущественной неизвестной старое обозначение x. Вхождения терма t в условия задачи Z заменяются на x; переменная y в утверждении R заменяется на x, и отличные от "число(x)" конъюнктивные члены данного утверждения заносятся в условия задачи Z.

Учет комментария "внимание"

Если какой-либо прием не срабатывает из-за того, что не удается убедиться в истинности некоторого утверждения A, то он может установить режим слежения за появлением этого утверждения при последующих преобразованиях задачи (например, при выводе следствий). Для этого создается комментарий (внимание S), список S которого перечисляет тройки (A T u). Здесь T - терм задачи (условие либо посылка), вес которого должен быть уменьшен до величины u при обнаружении утверждения A. Предполагается, что уменьшение веса вызовет повторную попытку применения приема, которая может оказаться уже успешной. Если комментарий относится к списку посылок задачи, то отслеживается только появление утверждения A в списке посылок; если он относится ко всей задаче, то - появление утверждения A в списке условий. Чтобы установка на слежение была введена, в описание приема на ГЕНОЛОГе добавляется указатель "См(i)", где i - номер антецедента теоремы, для которого вводится слежение. Заметим, что хотя режим использования комментариев "внимание" и был введен в процессе обучения, однако впоследствии надобность в нем отпала, и сейчас он не востребован.

Программы приемов, учитывающих комментарий "внимание", находятся через пункты "Учет комментария посылок ВНИМАНИЕ", "Учет комментария ВНИМАНИЕ" оглавления программ. Эти приемы просматривают все посылки либо условия задачи, имеющие вес 0, и сравнивают их с утверждениями, за которыми введено слежение. При совпадении реализуются уменьшение веса и отключение слежения за найденным утверждением.

Учет комментария "Случай"

Иногда разбор случаев в задаче нужно организовать по той причине, что этого требует какой-либо вспомогательный пакетный оператор. Необходимость разбора случаев усматривается в процессе применения последнего, однако прямых способов передать информацию о необходимости разбора случаев внешней задачи не предусмотрено - оператор просто оказывается ложным либо выдает отказ, и на этом все заканчивается. Поэтому для передачи информации приходится использовать комментарий (Случай A) к внешней задаче, создаваемый специальным приемом пакетного оператора. Здесь A - дизъюнкция, определяющая разбор случаев. Прием задачи на описание, учитывающий данный комментарий, срабатывает на любом уровне, как только этот комментарий появляется. Он исключает комментарий, регистрирует утверждение A в списке условий, и сопровождает его комментарием "разборслучаев". Прием используется крайне редко.

Усмотрение принадлежности элемента классу

В заключение рассмотрим несколько нетипичный прием. Хотя он и относится к частному логическому символу - описателю "класс", программу его пришлось закрепить за типом задачи "описать".

Если задача имеет посылку вида $\text{класс}_x(A(x)\ \&\ldots) = y$, где $A(x)$ - элементарное утверждение, содержащее все переменные связывающей приставки x, то при появлении посылок вида $A(p)$ предпринимается проверка истинности остальных утверждений под описателем "класс", после чего выводится следствие $p \in y$. Этот прием можно было бы реализовать на ГЕНОЛОГе и предпринимать попытки применения его при обнаружении символа "класс". Однако, тогда будут игнорироваться новые утверждения $A(p)$, появляющиеся после попытки применения приема. Так как априори ничего не известно о понятиях, которые могут встретиться в утверждениях $A(p)$, пришлось создать общий прием, активизируемый на символе "описать" при текущем уровне 4. Он просматривает элементарные посылки веса 4 - кандидаты на роль $A(p)$, ищет для них подходящее равенство, определяющее класс, выполняет необходимые проверки, и выводит следствие.

1.2 Приемы задач на преобразование

Учет о.д.з.

Ввод утверждений, формулирующих условия на о.д.з., для задач на преобразование происходит так же, как и для задач на описание. На текущем уровне 0 проверяется наличие цели "одз"; эта цель удаляется, и предпринимается обращение к процедуре "одз".

Создание комментариев, определяющих схему сопровождения по о.д.з.

Прием - такой же, как для задач на описание.

Идентичные посылки задачи

Прием, устраняющий дублирование в посылках, такой же, как в случае задач на описание.

Понижение веса посылки

При создании вспомогательной задачи на преобразование веса посылок часто берутся большими, чем максимальный уровень обращения (например, равными "двадцать"). Это нужно для ускорения решения, чтобы не тратить время на повторное рассмотрение посылок, образующих уже "знакомый" контекст какой-то внешней задачи. Однако, такое замораживание посылок все же приходится частично отменять, используя специальные приемы. В частности, нежелательно блокировать рассмотрение равенств $A = B$, у которых заменяемая часть A встречается в преобразуемом терме t. Они, будучи заблаговременно ориентированы нужным образом, определяют стандартизацию обозначений одного и того же объекта, так что решателю стоило бы заменить в t выражение A на B. Чтобы это произошло, используется прием, который по достижения максимального уровня просматривает посылки $A = B$, вес которых больше текущего уровня, причем A входит в условие. Веса таких посылок заменяются на 0, и реализуется повторное рассмотрение задачи.

Упрощение посылок

Прием, упрощающий посылки, аналогичен соответствующему приему задач на описание. Проверяется, что посылка не была получена при выводе следствий и ранее не упрощалась. Для ее упрощения используется вспомогательная задача на преобразование, решаемая с достаточно сильным ограничением на трудоемкость (всего 50000 шагов работы интерпретатора). Таким образом, реализуется лишь самое быстрое поверхностное упрощение посылок.

Завершающее редактирование ответа

По исчерпании средств, отпущенных для решения задачи на преобразование, предпринимается попытка компактной переформулировки ее условия с помощью процедуры "свертка". Она инициируется для текущего уровня, равного максимальному уровню. Предварительно анализируется целевая установка, чтобы не применять прием там, где он может нарушить требуемый вид ответа. Для блокировки повторного применения приема служит комментарий "длина". Если максимальный уровень задачи меньше 7, то прием применяется лишь при наличии цели "учетрезультата", т.е. при завершающем редактировании ответа задачи на описание, имеющей цель (известно ...).

Выдача ответа

Ответ задачи на преобразование выдается приемами, сгруппированными в разделе "Задачи на преобразование" - "Выдача ответа" оглавления программ. Обычно это происходит на максимальном уровне, при исчерпании отведенных для решения задачи средств; см. пункт "Выдача ответа задачи на преобразование - общий случай". Здесь проверяется, что условие задачи удовлетворяет всем требованиям, накладываемым на ответ целевой установкой. Прежде всего, отбрасываются случаи, когда условие содержит введенную при решении вспомогательную переменную x, указанную в комментарии (вспомпараметр x). Учет остальных целей происходит с помощью справочника "преобразовать", которому последовательно предъявляются тройки (текущая цель - задача - ее условие). Символом обращения к справочнику служит заголовок цели. Если ни одна из целей не блокирует выдачи ответа, то в качестве ответа берется условие задачи.

Заметим, что контроль пригодности условия выполняется в начале цикла сканирования, соответствующего максимальному уровню. Это означает, что в случае выдачи ответа другие приемы, которые могли бы сработать на максимальном уровне, применены не будут. Если нужно обеспечить применение каких-либо конкретных приемов, например, при упрощении выражения, то максимальный уровень обращения к задаче следует выбирать хотя бы на единицу большим их уровня срабатывания.

В специальных случаях ответ может быть выдан сразу же, на нулевом текущем уровне:

1. Если задача имеет цель (символ A), означающую немедленную выдачу ответа при получении заголовка A преобразуемого терма;

2. Если преобразуется не выражение, а утверждение, которое совпало с одной из посылок задачи. Тогда сразу выдается ответ "истина";

3. Если задача имеет цель (упрощение A), изначающую немедленную выдачу ответа при получении терма, длина которого меньше длины терма A;

4. Если задача имеет цель (заголовок $A_1 \ldots A_n$) и не имеет других целей (кроме, быть может, цели с заголовком "декомпозиция"), причем получен терм с заголовком A_i; $i \in \{1, \ldots, n\}$;

5. Если задача имеет цель "деление", означающую немедленную выдачу ответа при устранении дробного вида преобразуемого терма.

Заметим, что перечисленные особые случаи встречаются в работе системы исключительно редко.

Учет цели "заголовок"

Если задача имеет цель (заголовок $A_1 \dots A_n$), указывающую на необходимость преобразования условия к виду терма с заголовком - элементом списка $\{A_1, \dots, A_n\}$, то предпринимается попытка воспользоваться нормализаторами преобразования к заданным заголовкам. Эти нормализаторы находятся с помощью справочника "нормзаголовок". Прием применяется в процедурах, связанных с базой теорем.

Учет комментария посылок "внимание"

Прием - такой же, как для задачи на описание.

Приближенное вычисление константного выражения

Если задача имеет цель "выч", то ее условие представляет собой константный терм, и нужно найти его приближенное численное значение, ориентируясь на уровень точности, обеспечиваемый вычислениями с помощью математического сопроцессора в формате чисел "с плавающей запятой". В этой ситуации применяется либо процедура "вычпрог", составляющая программу вычислений, реализуемую далее оператором "Выч(программа ...)", либо процедура "вычконст", осуществляющая рекурсивное вычисление значения. Первая возможность несколько быстрее, и вторая используется только в случаях, когда условие содержит операции, не реализуемые командами программы оператора "Выч(программа ...)".

1.3 Приемы задач на доказательство

Учет о.д.з.

Выполняется так же, как в предыдущих случаях.

Создание комментариев, определяющих схему сопровождения по о.д.з.

Так же, как в предыдущих случаях.

Идентичные посылки задачи

Так же, как ранее.

Противоположные посылки задачи

Если при просмотре посылок A, имеющих вес 0, обнаруживается, что отрицание A тоже имеется в посылках, то ранее введенные комментарии (выводимо ...) удаляются; регистрируется информация о том, что ответ извлечен из найденных противоположных посылок, и выдается ответ "истина".

Совпадение условия с одной из посылок

Если на текущем уровне 0 усматривается, что условие совпадает с некоторой посылкой, то выдается ответ "истина". Предварительно в комментарии (выводимо ...) регистрируется информация об использовании этой посылки.

Отрицание конъюнктивного члена условия входит в посылки

Если условие задачи имеет вид $A_1\&\ldots\&A_n$, причем отрицание некоторого A_i содержится в посылках, то условие задачи заменяется на константу "ложь". Уровень срабатывания приема равен 0.

Упрощение посылок

Прием аналогичен приему, упрощающему посылки задачи на описание. Уровень применения его равен 2. После проверки того, что условие задачи не имеет заголовка "и" либо "существует" (иначе будут применяться другие, более приоритетные приемы) начинается просмотр посылок A. Отбираются те посылки, для которых целесообразно предпринимать попытку их упрощения. Она реализуется с помощью вспомогательной задачи на преобразование.

Доказательство от противного

Если прямое доказательство не удается довести до конца с помощью приемов малых уровней, то на уровне 5 либо 7 предпринимается попытка доказательства от противного. Уровень 5 относится к случаям, когда правдоподобно извлечение достаточно информативных следствий из отрицания условия A: либо A имеет вид "не(равно(x t))", где x - переменная, не входящая в t, либо A имеет вид равенства двух нечисловых переменных, либо A содержит выражение "значение(f t)"для константного t, а в посылках имеется кванторная импликация для f. В прочих случаях уровень срабатывания равен 7. Применение приема блокируется, если доказывается шаг индукции. Кроме того, оно блокируется, если A не имеет вида "не(равно(x t))", а задача содержит некоторые конкретные понятия ("расстояние", "угол", и т.п.).

Разумеется, все это - сугубо эвристические ограничители. Для рассмотренного обучающего материала они достаточны, так как отсекают большинство заведомо бесполезных попыток доказательства от противного. Однако, эти ограничители ни в коей мере не должны рассматриваться как нечто завершенное. Прием доказательства от противного - чрезвычайно общий метод, и для хорошего управления им понадобится накопление целой базы эвристических фильтров, учитывающих особенности конкретных ситуаций для различных предметных областей. Возможно, по мере развития этой базы придется распределить фильтры по программам вспомогательного справочника.

Если попытка доказательства от противного не отвергнута, то вводится комментарий "противоречие", блокирующий повторную попытку, и создается вспомогательная задача на исследование Z'. Ее список посылок образован всеми посылками задачи на доказательство, к которым добавлено отрицание условия A. Задача Z' имеет единственную цель "противоречие". Происходит обращение к решению задачи Z' с максимальным уровнем 7. Если после этого в посылках задачи Z' обнаруживается константа "ложь", то на текущую задачу Z выдается ответ "истина".

Использование замороженного равенства в посылках

Для ускоренной проверки часто используются задачи, у которых веса посылок больше, чем максимальный уровень. Обычно они сопровождаются комментарием "извлекается". Однако, у таких задач оказывается заблокированной стандартизация обозначений одного и того же объекта, использующая равенства из посылок. Это может привести к выдаче отказа даже в очевидных случаях. Поэтому предусмотрен специальный прием, который при наличии цели "извлекается" проверяет наличие посылок вида $A_1 = A_2$, где A_1 входит в условие задачи. Для этих посылок предпринимается замена в условии всех вхождений A_1 на A_2. Веса посылок не изменяются. Уровень срабатывания приема равен 1; для блокировки его повторного применения служит комментарий "результподст".

Упрощение условия

На достаточно высоком уровне сканирования (шестом) предпринимается попытка обратиться к вспомогательной задаче на преобразование для упрощения условия. Прием блокируется, если условие имеет своим заголовком конъюнкцию, дизъюнкцию либо квантор, либо не содержит более чем однобуквенных выражений. Имеется ряд других фильтров, ограничивающих применение приема. Для блокировки повторных попыток упрощения служит комментарий "упростить".

Развертка - свертка неэлементарных посылок

Если посылка задачи содержит кванторы и описатели, то иногда ее удается существенным образом упростить, используя сначала полную расшифровку входящих в нее понятий по определениям, а затем, после общей стандартизации результата расшифровки, обратную свертку "по определениям". Прием, выполняющий такие действия, активизируется на 6-м уровне. Для расшифровки "по определениям" служит обращение к вспомогательной задаче на описание, условие которой образовано выделенной неэлементарной посылкой, а цели суть "редакция", "развертка", "полный", "прямойответ". Она решается с небольшим уровнем обращения - 4. Свертка выполняется аналогичной задачей, в отсутствии цели "развертка". Для блокировки повторов служит комментарий к посылке "стандтерм".

Выбор новой неизвестной при получении известного условия

В некоторых предметных областях для доказательства утверждения удобно бывает выделить среди переменных подмножества условно "известных" и "неизвестных". Выделенные неизвестные регистрируются в комментарии к задаче (неизвестные $x_1 \ldots x_n$); известные - в комментарии (известно $a_1 \ldots a_m$). При этом начинают работать приемы, предпринимающие попытки выразить "неизвестные" через "известные". После подстановки таких выражений в условие задачи оно обычно становится очевидным. Данная схема применяется, например, при доказательстве геометрических соотношений. В качестве неизвестной здесь естественно выбирать какую-либо числовую переменную, входящую в условие, а в качестве известных - прочие числовые переменные. Если в процессе преобразований (например, выражения неизвестной через известные) условие перестает содержать неизвестные, то выбирается новая неизвестная. Прием, выполняющий такой выбор, активизируется на уровне 7. В качестве новой неизвестной он выбирает произвольный параметр, указанный в комментарии (известно ...) и входящий в условие. Этот параметр переносится из списка (известно ...) в список (неизвестные ...). Веса посылок и условия при этом понижаются до 0.

Учет комментария посылок "внимание"

То же, что для задач на описание и преобразование.

Ввод комментария "раздел"

Для различных целей может понадобиться список максимальных разделов, к которым относятся встречающиеся в задаче понятия. Такой список A создается в начале решения задачи (при текущем уровне 0) и сохраняется в комментарии (раздел A).

1.4 Приемы задач на исследование

Учет о.д.з.

Выполняется так же, как в предыдущих случаях.

Создание комментариев, определяющих схему сопровождения по о.д.з.

Так же, как в предыдущих случаях.

Идентичные посылки задачи

Так же, как ранее.

Противоположные посылки задачи

Если некоторая посылка задачи, имеющая вес 0, оказывается совпадающей с отрицанием другой посылки, то она заменяется на константу "ложь".

Удаление чрезмерно больших посылок

Чрезмерно громоздкие посылки, возникающие при выводе следствий, чаще всего оказываются бесполезны для решения задачи. При этом они сильно замедляют действия системы. Поэтому желательно иметь прием, который бы исключал такие посылки. Однако, необходимо блокировать отбрасывание посылок, несущих уникальную и необходимую в задаче информацию. Например, не следует удалять посылки вида $x = t$, где x - неизвестная внешней задачи на описание, t - известное выражение. Заметим, что иногда эти посылки и оказываются рекордно длинными, например, если ответ по необходимости громоздок. Здесь возникает непростая проблема накопления эвристических ограничителей приема. На текущий момент накоплена какая-то, видимо, далеко не идеальная система таких ограничителей. Подробнее можно ознакомиться с ними по программе приема (см. пункт "Удаление чрезмерно больших посылок в задачах, имеющих цель ИЗВЕСТНО" оглавления программ). Посылка считается чрезмерно длинной, если в ней более 130 символов. В специальных случаях эта константа увеличивается до 300. Блокируется отбрасывание дизъюнкций, снабженных комментарием "разборслучаев", а также конъюнкций (впрочем, последнее отменяется, если неизвестные задачи частично определены). Блокируется также отбрасывание посылок, возникших с самого начала решения задачи.

Усмотрение ответа внешней задачи на описание

При решении задачи на исследование Z', являющейся блоком анализа задачи на описание Z, предпринимается попытка получить такую систему следствий, которая явно указала бы искомые значения неизвестных, и таким образом могла бы быть взята (с необходимыми коррекциями) в качестве ответа задачи Z. Усмотрение того, что данная система следствий или ее фрагмент уже получены, обычно выполняется приемами символа "равно". Однако, в одном специальном случае для усмотрения ответа понадобился общий прием задач на исследование. Этот случай связан с решением геометрических задач на построение или похожих на них задач. Процедура решения заключается в том, что предпринимается некоторая изначальная фиксация нескольких точек как "известных" - для устранения степеней свободы, допускаемых параллельными сдвигами, поворотами и т.п. Далее происходит анализ чертежа, в процессе которого постепенно доопределяются (с помощью допустимых средств построения) прочие точки. На каждом шаге доопределения возникают одно или несколько утверждений, рассматриваемых как выражение новой точки через старые. После этого данная точка уже считается известной. Утверждения, определяющие фиксацию точки A, помечаются комментариями (найдено A). В некоторый момент оказывается, что все неизвестные в задаче Z точки определились. Тогда и должен сработать указанный выше прием усмотрения ответа. Уровень срабатывания его равен 1; программу можно найти в пункте "Попытка усмотрения ответа внешней задачи на описание при определении всех ее неизвестных" оглавления программ.

Прежде всего, прием анализирует целевую установку и проверяет непересечение списков неизвестных задач Z и Z'. Последнее означает, что все неизвестные задачи Z определены. Составляется список S пар (переменная - утверждения, определяющие ее значение согласно комментариям "найдено"). Здесь могут встречаться не только неизвестные задачи Z, но и вспомогательные переменные, через которые будут выражаться эти неизвестные. В программе список S присваивается переменной x7. Пары списка S переупорядочиваются так, чтобы новые переменные определялись только с использованием ранее определенных; это дает список S'. В программе он присвоен переменной x8. Далее по S' находится список R всех утверждений, используемых, прямо или косвенно, для определения неизвестных задачи Z. В программе список R присвоен переменной x9. Составляется список P всех известных посылок задачи Z', не содержащих переменных, определенных списком S, пополненный всеми известными условиями задачи Z. После этого предпринимается попытка доказать, что условия задачи Z являются следствиями ее посылок, объединенных со списками R и P. Для задач на построение этот шаг означает попытку доказать, что найденная при анализе чертежа последовательность действий приводит к желаемым результатам. Использованные при доказательстве утверждения списка P добавляются, после упрощения их вспомогательными задачами на описание, к списку R. Последний и рассматривается как ответ. Чтобы зарегистрировать его в таком качестве, прием предпринимает замену всех условий внешней задачи Z на утверждения списка R, а список неизвестных задачи Z пополняет теми вспомогательными переменными, которые вошли в R. Затем создается комментарий "ответ" к задаче Z, и происходит возвращение к ней от задачи Z'.

Учет комментария посылок "внимание"

То же, что для задач на описание, преобразование и доказательство.

Пополнение накопителя "полныепосылки"

Если задача на описание имеет своим условием кванторную импликацию $\forall_x(A_1 \,\&\ldots\,\&\, A_n \to A_0)$, то обычно предпринимается попытка найти ее бескванторную эквивалентную переформулировку. В простейших случаях этого удается добиться с помощью кванторных определений либо за счет явного разрешения подкванторных утверждений относительно переменных связывающей приставки x. В более сложных случаях приходится применять процедуру накопления такого списка бескванторных следствий кванторной импликации, которые оказались бы достаточны для истинности этой импликации. Это - хорошо известная процедура усиления необходимого условия, пока оно не станет достаточным. Для ее реализации создан специальный прием символа "длялюбого", который будет подробнее описан впоследствии. Вывод следствий осуществляется в рамках вспомогательной задачи на исследование, имеющей цель "длялюбого". Этой задаче передаются в качестве посылок все антецеденты A_i и все конъюнктивные члены консеквента A_0. Предварительно проверяется реализуемость антецедентов.

Накопление бескванторных следствий кванторной импликации происходит в комментарии (полныепосылки S). Его обеспечивает общий прием задач на исследование, просматривающий вновь выведенные посылки P. Если такая посылка не содержит переменных связывающей приставки x, то она непосредственно регистрируется в S. Иначе - рассматривается кванторная импликация $\forall_x(A_1 \,\&\ldots\,\&\, A_n \to P)$, и предпринимается попытка преобразовать ее в бескванторное утверждение с помощью вспомогательной задачи на описание. Если это удается, результат тоже регистрируется в списке S.

1.5 Приемы символа "и"

Переходим к приемам, активизируемым при просмотре входящих в условия и посылки задачи логических символов. Эти приемы, в основном делятся на три группы - простые общелогические упрощения утверждений, применяемые почти без ограничений; приемы, планирующие общий ход действий по решению задачи (разбор случаев; исключение неизвестной, явно выраженной через другие неизвестные, и т.п.), и приемы, осуществляющие вывод следствий. Некоторые приемы имеют достаточно общий характер, в то время как оценить из общих соображений целесообразность их применения бывает затруднительно. Тогда приходится прибегать к специальным "решающим" справочникам, подлежащим пополнению в процессе обучения системы и оценивающим целесообразность применения данного общего приема в зависимости от тех понятий, которые встретились в конкретном контексте его срабатывания.

Начнем с символа "и". Группу его реализованных на ЛОСе приемов можно найти в разделе оглавления программ "Приемы решателя" - "Общие приемы" - "Конъюнкция". Все эти приемы крайне просты.

Конъюнкция с противоположными членами

Прием применяется на уровне 0. Он производит попарные сравнения различных операндов конъюнкции. Если обнаруживаются операнды, один из которых является отрицанием другого, то конъюнкция заменяется на константу "ложь".

Конъюнкция с повторяющимися членами

Прием аналогичен предыдущему, но проверяет совпадение различных операндов. При обнаружении совпадения один из операндов отбрасывается. В случае, когда операндов было всего два, отбрасывается также символ конъюнкции.

Конъюнктивное условие задачи на доказательство

Если условие задачи на доказательство имеет вид $A_1 \& \ldots \& A_n$, то обычно эта задача разбивается на n независимо решаемых задач, имеющих условия $A_1, \ldots, A_n$. Уровень срабатывания приема равен 2 - чтобы в простейших случаях успели сработать приемы, устраняющие конъюнкцию прочими средствами. Блокировка приема была ведена лишь для геометрических задач на доказательство, которые оказалось целесообразнее решать без разбиения условия. Комментарии исходной задачи передаются подзадачам после обработки их справочником "конъюнктчлен". Список посылок каждой подзадачи представляет собой независимую копию списка посылок исходной задачи. Информация об использовании посылок извлекается из каждой подзадачи после ее решения, и объединенный список передается исходной задаче перед выдачей ответа "истина" на нее.

Конъюнктивное условие задачи на описание

Если задача на описание имеет своим условием утверждение $A_1 \& \ldots \& A_n$, то это условие заменяется на A_1, а утверждения $A_2, \ldots, A_n$ присоединяются в качестве ноых условий. Комментарии к исходному условию передаются измененному и новым условиям после обработки их справочником "и". Веса этих условий полагаются равными 0. Прием применяется без ограничений, и уровень срабатывания его равен 0.

Конъюнктивная посылка

Если утверждение $A_1 \& \ldots \& A_n$ представляет собой посылку задачи, то она разбивается на отдельные конъюнктивные члены. Это происходит так же, как в предыдущем приеме, причем используется тот же самый справочник "и". Для задач на исследование, имеющих цель "текстоваязадача", работа приема несколько модифицирована - посылки $A_2, \ldots, A_n$ добавляются не в начале списка посылок, как в общем случае, а непосредственно после измененной посылки. Это делается для сохранения той информации, которая определяется порядком слов в анализируемом тексте, например, информации о временной последовательности событий. Впрочем, для сохранения последней предусмотрены и другие средства.

1.6 Приемы символа "или"

Количество приемов в этом разделе значительно больше, чем в предыдущем, и некоторые из них уже достаточно сложны.

Простейшая стандартизация

Начнем с серии простых приемов, осуществляющих упрощение дизъюнкции.

1. Устранение повторных дизъюнктивных членов. Если какие-то два операнда A_i, A_j дизъюнкции $A_1 \vee \ldots \vee A_n$ совпадают, то один из них удаляется. При $n = 2$ удаляется также знак дизъюнкции. Уровень срабатывания приема равен 0.

2. Дизъюнкция с противоположными членами. Если какой-либо операнд дизъюнкции совпадает с отрицанием другого операнда, то дизъюнкция заменяется на константу "истина". Исключение составляют дизъюнктивные условия и посылки, снабженные комментарием "разборслучаев". Обычно они имеют противоположные операнды, но исключать их не следует, так как дизъюнкция должна инициировать срабатывание приема, выполняющего разбор случаев. Уровень срабатывания приема равен 0.

3. Преобразование дизъюнкции, если в контексте имеется дизъюнктивный член или его отрицание. Если в контексте, относительно которого рассматривается дизъюнкция $A_1 \vee \ldots \vee A_n$, имеется утверждение A_i, то эта дизъюнкция заменяется на константу "истина". Если в контексте имеется отрицание утверждения A_i, то дизъюнкция заменяется на $A_1 \vee \ldots \ A_{i-1} \vee A_{i+1} \vee \ldots \vee A_n$. Уровни срабатывания приема - 0 либо 7. Уровень 7 введен для повторного контроля, так как возможность применения приема зависит от появления в контексте новых утверждений.

4. Анализ конъюнктивных членов дизъюнктивного члена. Если дизъюнктивный член A_i имеет вид $B_1 \ \& \ldots \& B_m$, причем некоторое B_i встречается в контексте дизъюнкции, то оно исключается из A_i. Если в контексте встречается отрицание утверждения B_i, то исключается дизъюнктивный член A_i. Оба приема срабатывают на уровне 0.

5. Усмотрение в контексте дизъюнкции, поглощающей данную. Если в контексте имеется дизъюнкция, множество дизъюнктивных членов которой является подмножеством множества $A_1, \ldots, A_n$ дизъюнктивных членов текущей дизъюнкции, то последняя заменяется на константу "истина". Уровни срабатывания приема равны 0 либо 7.

6. Вынесение за скобку общей части двух дизъюнктивных членов. Если два дизъюнктивных члена A_i, A_j представляют собой конъюнкции, имеющие непустое множество одинаковых членов, т.е. представимые в виде $B\&C$ и $B\&D$, то они объединяются в один новый дизъюнктивный член $B\&(C \vee D)$. Имеются небольшие ограничения на применение данного приема - исключение составляют случай задачи на преобразование, имеющей цель "днф", а также случай редактирования ответа задачи на описание, когда C содержит явное выражение некоторой неизвестной, встречающейся в B. Уровень срабатывания приема равен 1.

7. Поглощение дизъюнктивного члена. Если некоторый дизъюнктивный член A_i имеет своим конъюнктивным членом дизъюнктивный член A_j, то A_i отбрасывается. Уровень срабатывания приема равен 1.

8. Использование оператора "блокдизъюнкции" для упрощения дизъюнктивного условия в редактируемом ответе задачи на описание. Оператор "блокдизъюнкции" включает в себя три простейших преобразования: исключение вложенных дизъюнкций и вложенных конъюнкций, а также вынесение за скобку общей части двух дизъюнктивных членов. Прием применяется на текущем уровне 0. Для блокировки повторных попыток обработки им того же самого условия используется комментарий "блокдизъюнкции".
9. Использование преобразования $A \vee (B\&(A \vee C)) = A \vee (B\&C)$. Прием применяется на уровне 1.

Как легко заметить, предложенный список упрощающих дизъюнкцию приемов не претендует на полноту. Однако, увеличение этого списка из каких-либо общих соображений, в особенности для малых уровней срабатывания, могло бы повлечь за собой неоправданное замедление работы решателя. Поэтому данная коллекция включаетв себя лишь то, что фактически было востребовано на проработанном обучающем материале, и дальнейшее пополнение ее должно происходить с учетом целесообразности временных затрат решателя на попытки применения новых приемов.

Дизъюнктивное условие задачи на доказательство

Если задача на доказательство имеет условие $A_1 \vee \ldots \vee A_n$, то она сводится к задаче на доказательство какого-то одного утверждения A_i, с присоединением к посылкам отрицаний утверждений $A_1, \ldots, A_{i-1}, A_{i+1}, \ldots, A_n$. Чтобы выбрать нужное A_i, предварительно выполняется переупорядочение дизъюнктивных членов: во-первых, предпочтение отдается тому из двух утверждений, которое не имеет заголовка "существует"; во-вторых, тому, которое не имеет заголовка "не"; в-третьих, более короткому. Комментарии во вспомогательную задачу переносятся после обработки их справочником "или". Если на нее получен ответ "истина", то во внешнюю задачу переносится информация об использованных посылках, и затем выдается ответ "истина". Уровень срабатывания приема выбран равным 2, чтобы в случаях, когда это возможно, успевали сработать приемы общей стандартизации, исключающие дизъюнкцию.

Разбор случаев по дизъюнктивному условию задачи на описание

Это один из наиболее важных общелогических приемов. Уровни, на которых может быть предпринятая попытка его применения, равны 1,2,3,5 и 7. Конкретное значение зависит от контекста. Если дизъюнктивное условие появляется на этапе редактирования ответа задачи на описание (т.е. при решении задачи на описание, имеющей цель "редакция"), то разбор случаев допускается в двух ситуациях - либо дизъюнкция не содержит неизвестных и помечена комментарием "разборслучаев", либо в каком-либо из дизъюнктивных членов явно указывается значение несущественной неизвестной. В начале программы приема предусмотрена еще одна проверка того, что какой-либо дизъюнктивный член совпадает с условием задачи. Если это так, то дизъюнкция заменяется на константу "истина". Хотя данная проверка уже предпринималась на уровне 0 другим приемом, но с тех пор (по достижении текущего уровня) могли измениться прочие условия задачи, и повторение ее иногда оказывается результативным.

Далее уточняется уровень срабатывания приема. Если условие выделено комментарием "разборслучаев", то он равен 1. Для условия, содержащего неизвестные, уровень обычно равен 3. Если неизвестных в дизъюнкции нет, то уровень берется равным 7 при редактировании ответа задачи на описание и 5 в противном случае. При наличии цели "пример" ограничения на уровень срабатывания снимаются.

Чтобы блокировать повторную попытку применения данного приема при данном уровне срабатывания U, используется комментарий к условию (или U). Если имеется несколько дизъюнктивных условий, то предпочтение отдается тому, которое в каждом своем подслучае дает явное значение для какой-либо неизвестной задачи.

Определяется максимальный уровень, с которым будут решаться задачи для отдельных подслучаев (см. переменную x7 в программе приема). Обычно он равен максимальному уровню текущей задачи, хотя при решении задачи на подбор примера возможны попытки рассмотрения подслучаев с малым уровнем, равным 3. Если решается задача на подбор примера, то из двух подслучаев сначала рассматривается тот, который задается более коротким утверждением.

Далее находится список $A_1, \ldots, A_n$ дизъюнктивных членов рассматриваемого условия, и начинается их последовательный просмотр. Пусть A_i - текущий просматриваемый подслучай. Если задача - на подбор примера и имеет цель (независит ...), причем A_i содержит какую-либо переменную данной цели, то рассмотрение этого подслучая отменяется. Если выбранное дизъюнктивное условие было помечено комментарием "Одз", то предпринимается попытка скорректировать его члены (с сохранением эквивалентности исходному условию), присоединив к ним те необходимые для сопровождения по о.д.з. утверждения, которые можно усмотреть из отрицания остальных членов.

Наконец, создается вспомогательная задача Z', полученная из текущей задачи Z заменой дизъюнктивного условия на утверждение A_i. Здесь используется оператор "спуск", заменяющий заданное условие на новое, а также копирующий блок анализа и корректирующий комментарии. Если дизъюнктивное условие не имело комментария "разборслучаев", то вхождения в условия задачи Z' ранее разобранных подслучаев заменяются на константу "ложь". Корректируется блок анализа задачи Z'. Из него удаляется рассматриваемое дизъюнктивное условие и понижаются до 0 веса ряда посылок - например, тех, которые имеют общую переменную с A_i. Далее предпринимается обращение к решению задачи Z'. Если на нее получен "отказ", а задача Z не имела цели "пример" и имела цель "полный", то на задачу Z тоже выдается "отказ". Если на задачу Z' получен ответ R_i, отличный от символа "отказ", то прежде всего предпринимается перенесение из списка посылок задачи Z' в список посылок задачи Z тех утверждений, которые могут впоследствии оказаться полезными - они помечаются в задаче Z' комментариями "облвхожд". Аналогичная работа проводится с комментариями - справочник "преобразование" отбирает и корректирует их для перенесения из Z' в Z. Если R_i - константа "истина", то она сразу выдается в качестве ответа. Для константы "ложь" выделен один специальный подслучай, когда ее тоже можно сразу же выдать в качестве ответа. В прочих ситуациях частичный ответ R_i регистируется в накопителе ответа (его роль играет программная переменная x8). Для задач на подбор примера определяются условия Q_i на известные параметры, при которых R_i дает требуемый пример. Проверяется, не является ли дизъюнкция $Q_1, \ldots, Q_i$ следствием посылок задачи Z. Если это так, то уже можно выдавать полный ответ, и рассмотрение последующих подслучаев A_j обрывается.

По окончании рассмотрения всех A_i (с учетом указанной выше возможности обрыва) возникают списки ответов $R_1, \ldots, R_k$ для разобранных подслучаев. Если решается задача на поиск примера, то возникает также список условий $Q_1, \ldots, Q_k$ на известные параметры, относительно которых найдены соответствующие ответы. Первый список (в виде наборов конъюнктивных членов утверждений R_i) присвоен переменной x8; второй - переменной x9. Переменная x10 является индикатором охвата условиями $Q_1, \ldots, Q_k$ всего многообразия возможностей, предоставляемых посылками задачи Z: если все случаи охвачены, то она равна 0.

Прежде всего, в ситуации с задачей на поиск примера и при ненулевом x10, предпринимается попытка установить, что дизъюнкция утверждений $Q_1, \ldots, Q_k$ является следствием посылок задачи Z. Если это не так, то применение приема блокируется. Иначе - определяются общая не содержащая неизвестных часть M всех ответов $R_1, \ldots, R_k$, а также остатки $S_1, \ldots, S_k$ этих ответов. Определяется утверждение R вида $M\ \&(S_1 \vee \ldots \vee S_k))$, в котором выполняются тривиальные упрощения, соответствующие тождественной истинности или ложности каких - либо S_i. Если задача Z не имеет цели "упростить", то в качестве ответа на нее выдается R. Это же происходит еще в некоторых специальных случаях. Иначе - предпринимается обращение к нормализатору "нормили", который и дает окончательный ответ на задачу.

При данной схеме разбора случаев получается, что все завершающее редактирование дизъюнкции ответов, найденных в подслучаях, происходит только внутри нормализатора "нормили". Никаких преобразований этой дизъюнкции в рамках сканирования той или иной вспомогательной задачи не предусмотрено. Соответственно, нормализатор "нормили" оказывается исключительно большой процедурой, вбирающей в себя всю технику склейки дизъюнктивных логических конструкций, имеющуюся в различных предметных областях. Подробнее его содержимое будет рассмотрено ниже: общелогическая составляющая в данной главе, остальное - в главах, посвященных отдельным предметным областям.

Разбор случаев по дизъюнктивной посылке задачи на описание

Если задача на описание имеет своей посылкой дизъюнкцию $A_1 \vee \ldots \vee A_n$, то она может решаться путем разбора случаев по этой дизъюнкции. Уровни срабатывания приема, реализующего разбор случаев, равны 1, 2 либо 3. Уровень равен 1, если посылка выделена комментарием "разборслучаев"; равен 2, если задача имеет цель "пример", и равен 3 в прочих случаях. Если посылка не выделена комментарием "разборслучаев", то проверяется, что хотя бы один ее параметр входит в условия задачи. Далее начинается просмотр утверждений, определяющих подслучаи. Пусть A_i - текущее такое утверждение. Создается вспомогательная задача Z', полученная из Z заменой рассматриваемой посылки на A_i. В программе эта задача присвоена переменной x11. Комментарии к посылкам переносятся в Z' из Z при помощи справочника "вариант"; комментарии к условиям - при помощи справочника "компонента". Добавляется цель "текпосылка", имеющая чисто информационный характер - она означает, что данная вспомогательная задача на описание возникла при разборе случаев по дизъюнктивной посылке. Если задача Z имела блок анализа, то копия его передается задаче Z' после добавления посылки A_i. Веса посылок задачи Z', которые содержат условные выражения, имеющие своим подутверждением утверждение A_i с отброшенным внешним отрицанием, уменьшаются до 0.

Далее происходит обращение к решению задачи Z' с тем же максимальным уровнем, какой имела задача Z. Если получается "отказ", то и на задачу Z выдается "отказ". Иначе - по завершении рассмотрения всех подслучаев возникает набор ответов $R_1, \ldots, R_n$. Если задача Z имеет цель (независит $\ldots$) и не имеет цели "полный" (что обычно выполняется при наличии цели "независит"), то в качестве ответа берется конъюнкция всех R_i. Быть может, в каких-то случаях это существенно сужает ответ, но зато исключает его зависимость от указанных в цели параметров. При отсутствии цели (независит $\ldots$) в качестве заготовки ответа рассматривается утверждение вида $(A_1 \& R_1) \vee \ldots \vee (A_n \& R_n)$. Оно упрощается с помошью вспомогательной задачи на описание, цели которой получены добавлением к целям задачи Z символа "редакция", и затем выдается как окончательный ответ.

Разбор случаев по дизъюнктивной посылке задачи на доказательство или на преобразование

Разбор случаев по дизъюнктивной посылке задачи на преобразование предпринимается лишь тогда, когда эта посылка выделена комментарием "разборслучаев". Уровень срабатывания приема при этом равен 0. Для задачи на доказательство уровень срабатывания равен 1 при наличии комментария посылки "разборслучаев", равен 2, если хотя бы одна из альтернатив не имеет вида отрицания равенства, и равен 5 в остальных случаях. Для задачи на преобразование, результат разбора случаев не всегда является окончательным ответом. В некоторых ситуациях он лишь замещает текущую версию преобразуемого условия. Тогда, с целью блокировки повторных применений приема, вводится комментарий посылки "или".

Обозначим для дальнейших рассмотрений дизъюнктивную посылку через $A_1 \vee \ldots \vee A_n$. Если решается задача на преобразование, имеющая цели (обозначение $\ldots$), то составляется список S всех переменных, упоминаемых в таких целях. Эти переменные суть вспомогательные обозначения, которые не должны входить в ответ задачи. Поэтому нежелательно их появление в альтернативах A_i, которые войдут в условные выражения для результата разбора случаев. Чтобы исключить такую возможность, рассматривается список "укороченных" альтернатив B_i, полученных из A_i отбрасыванием всех конъюнктивных членов, зависящих от S. Если все утверждения B_i отличны от константы "истина", причем усматривается их несовместность, то они заменяют в дальнейших преобразованиях утверждения A_i. Иначе применение приема блокируется.

Далее начинается последовательный просмотр дизъюнктивных членов посылки. Пусть A_i - текущий такой член. Вводится вспомогательная задача Z', полученная из текущей задачи Z заменой дизъюнктивной посылки на A_i. Если Z имеет комментарий "смпосылка", то веса всех посылок задачи Z' обнуливаются. Иначе на ноль заменяется только вес посылки A_i. Комментарии к посылкам и к задаче переносятся из Z в Z', соответственно, с помощью справочников "дизъюнктчлен" и "разборслучаев". Предпринимается понижение до нуля весов посылок, имеющих такие условные выражения, которые содержат A_i. После этого происходит обращение к решению задачи Z' с тем же максимальным уровнем, что и у задачи Z. Если получен "отказ", то в случае задачи на доказательство сразу выдается итоговый отказ. В случае задачи на преобразование получение отказа приводит лишь к обрыву попытки применения данного приема. Для задачи на преобразование заполняется накопитель частичных ответов $R_1, \ldots, R_n$, найденных в отдельных подслучаях. Значением программной переменной x6 становится пара, первым элементом которой служит данный накопитель, а вторым - накопитель использованных во всех подслучаях посылок.

По окончании разбора случаев в задаче на доказательство выдается ответ "истина". Предварительно обеспечивается коррекция комментария (выводимо $\ldots$) к текущей задаче Z, чтобы он ссылался на все использованные при разборе подслучаев исходные посылки.

Если решалась задача на преобразование, то после получения списка частичных ответов $(R_1, \ldots, R_n)$ предпринимается попытка усмотреть возможность поглощения одних подслучаев другими. Пусть, например, некоторое утверждение A_i имело своими конъюнктивными членами равенства $x_1 = t_1, \ldots, x_m = t_m$, фиксирующие значения некоторых переменных $x_1, \ldots, x_m$. Если в другом утверждении A_j значения этих переменных не были фиксированы, причем удается усмотреть равенство выражения R_i результату подстановки выражений $t_1, \ldots, t_m$ в R_j вместо переменных $x_1, \ldots, x_m$, предполагая истинность A_i, то вместо R_i в подслучае A_i можно использовать R_j.

Далее строится условное выражение R вида "вариант(A_1 R_1 вариант(A_2 R_2 ... вариант(A_{n-1} R_{n-1} R_n) ...))". Если какие-либо подслучаи A_i в действительности оказались невозможными и в качестве R_i был получен символ "противоречие", то эти подслучаи при построении R отбрасываются. Чтобы получить окончательный результат, к R применяется нормализатор "упрощвариант". Исключение составляют задачи, имеющие цель "класс". Тогда нормализация отменяется, и вместо выдачи ответа выражение R замещает условие текущей задачи Z. Как и в случае задач на доказательство, предпринимается коррекция комментария (выводимо ...).

Преобразования дизъюнкции при редактировании ответа задачи на описание

При редактировании ответа задачи на описание могут встречаться дизъюнктивные условия, по которым не предпринимается разбор случаев. Для работы с ними предусмотрен ряд простых приемов, которые перечисляем ниже. Напомним, что этап редактирования ответа выделяется наличием цели "редакция".

1. Попытка усмотрения избыточности дизъюнкции отрицаний равенств. Если при редактировании ответа возникает условие вида $x_1 \neq t_1 \vee \ldots \vee x_n \neq t_n$, где переменные $x_1, \ldots, x_n$ не встречаются в термах $t_1, \ldots, t_n$, то просматриваются другие условия g, все свободные переменные которых содержатся в списке $x_1, \ldots, x_n$. Предпринимается попытка усмотреть из контекста утверждения g отрицание результата подстановки в g выражений $t_1, \ldots, x_n$ вместо переменных $x_1, \ldots, x_n$. Как только это удается, дизъюнктивное условие отбрасывается. Уровень срабатывания приема равен 2.

2. Перемещение вглубь дизъюнкции тех утверждений, для которых внутри дизъюнкции явным образом определяются значения их неизвестных. Если в условии задачи на описание, имеющей цель "редакция", встречается дизъюнкция D вида $(x = t)\&A \vee B_1 \ldots \vee B_n$, где x - неизвестная, то рассматриваются все расположенные над ней дизъюнкции D' того же условия (допуская совпадение D' и D). Пусть $D' = C_1 \vee \ldots \vee C_m$. Если D' является операндом конъюнкции K, причем существуют другие ее операнды, зависящие от x, то рассматривается конъюнкция K' всех таких операндов. Они исключаются из K и переносятся в D', которая приобретает вид $C_1\&K' \vee \ldots \vee C_m\&K'$. Аналогичное преобразование выполняется, если D' - корневая дизъюнкция. Вместо операндов конъюнкции K тогда берутся все прочие условия задачи. Данный прием подготавливает возможность подстановки значения t вместо x в содержащие x утверждения. Он имеет два уровня срабатывания - 1 и 7.

3. Логическое упрощение. Если условие редактируемого ответа содержит дизъюнкцию вида $A \vee (B \& \neg A)$, где утверждение A содержит неизвестную, то эта дизъюнкция заменяется на $A \vee B$. Заметим, что в случае известного A данное преобразование уже нежелательно, так как устраняет явное указание на альтернативы, относительно которых даются фрагменты ответа. Еще один пример логической перегруппировки, выполняемой из соображений вынесения за скобки утверждений, не содержащих неизвестных, - использование тождества $(A \& B) \vee (C \& ((A \& D) \vee E)) = (A \& (B \vee (C \& D))) \vee (C \& E)$. Здесь A - утверждение без неизвестных; C - утверждение, все конъюнктивные члены которого содержат неизвестные. Первый прием срабатывает на уровнях 1 и 7; второй - на уровне 2. Видимо, они не исчерпывают логических перегруппировок в дизъюнкциях, которые могли бы быть полезными при редактировании ответа. Появление именно этих преобразований в решателе объясняется просто тем, что они оказались необходимы в рассмотренных к текущему моменту примерах.

Склейка двух дизъюнкций с общими членами

Если две дизъюнкции $A \vee B$ и $A \vee C$ являются операндами некоторой конъюнкции, то они объединяются в одну дизъюнкцию $A \vee (B \& C)$. Здесь A включает в себя все общие члены дизъюнкций. Уровень срабатывания приема равен 2. Он применяется без каких-либо ограничений, как в условиях, так и в посылках задач.

Вынесение за знак дизъюнкции конъюнктивного члена одного из операндов

Если решается задача на описание с целью "и", то ответ ее должен иметь вид конъюнкции элементарных утверждений. В этом случае предпринимается попытка извлечения из дизъюнкции $A_1 \vee \ldots \vee A_n$ конъюнктивных составляющих. Предварительно проверяется, что задача имеет цель "редакция" и рассматриваемое ее условие, содержащее дизъюнкцию, не имеет неизвестных. Затем просматриваются утверждения A_i. Для текущего такого утверждения просматриваются все его конъюнктивные члены B, представляющие собой элементарные утверждения. Если существует пакетный проверочный оператор, который мог бы распознавать истинность B, то он применяется для усмотрения B в каждой из оставшихся альтернатив A_j. В случае успеха дизъюнкция заменяется на конъюнкцию утверждения B и остаточной дизъюнкции $A'_1 \vee \ldots \vee A'_n$, где A'_j получено из A_j отбрасыванием конъюнктивного члена B, если таковой имелся. Уровень срабатывания приема равен 4.

Раскрывание скобок в условии задачи на преобразование, имеющей цель "днф."

Если задача на преобразование имеет цель "днф", т.е. ее условием является некоторое утверждение, которое надо преобразовать к виду дизъюнктивной нормальной формы относительно элементарных подутверждений, то применяется преобразование $A \& (B \vee C) = (A \& B) \vee (A \& C)$. Уровень срабатывания приема равен 2.

Дизъюнктивное условие, определяющее в одном из подслучаев явное значение известного параметра

Если условие задачи на описание не содержит неизвестных и имеет вид $(x = t \& A) \vee B_1 \vee \ldots \vee B_n$, где x - переменная, не входящая в терм t, то оно снабжается комментарием "разборслучаев". Вес условия при этом понижается до 0. Прием срабатывает на уровне 1.

Выход из блока анализа для разбора случаев по выведенной дизъюнкции

Если при выводе следствий в блоке анализа Z некоторой задачи на описание Z' возникает дизъюнкция $A_1 \vee \ldots \vee A_n$, содержащая неизвестные либо помеченная комментарием "разборслучаев", то на уровне 2 она инициирует разбор случаев. Предварительно проверяется отсутствие комментариев, блокирующих это действие, а также отсутствие в списке условий задачи Z' утверждений A_i либо дизъюнкции, полученной из данной отбрасыванием части членов.

Чтобы выполнить разбор случаев, сначала происходит перенесение из Z в Z' указанной дизъюнкции, а также, быть может, части сопровождающих ее посылок. Это обеспечивается процедурой "замещениеусловий", которая предпринимает попытку дополнить переносимые из Z в Z' утверждения таким образом, чтобы стало возможным отбросить какие-либо старые условия задачи Z', являющиеся следствиями новых. Процедура используется различными приемами, выполняющими перенесение во внешнюю задачу ответа, полученного в блоке анализа. Она будет описана ниже в специальном подразделе. Перенесенная в Z' дизъюнкция снабжается комментарием "разборслучаев". Затем прием осуществляет выход из блока анализа Z, используя для этого оператор "ответ(Z)". Дальнейшие действия будут выполняться другими приемами. Произойдут разбор случаев по дизъюнктивному условию задачи на описание и переходы к рассмотрению блока анализа в подслучаях A_i, так что прерванный вывод следствий будет продолжен для каждого такого подслучая по отдельности.

Если задача на исследование имеет цель "исследовать", то разбор случаев по выведенной дизъюнкции, снабженной комментарием "разборслучаев", инициируется быстрее - уже на нулевом уровне.

Нормализатор "нормили"

Основная часть приемов нормализатора "нормили" реализована на ГЕНОЛОГе. На ЛОСе реализованы лишь три второстепенных и притом весьма простых приема, которые и приводятся ниже.

1. Устранение отрицания равенства, имеющегося в некотором дизъюнктивном члене. Прием аналогичен приведенному выше приему сканирования задачи. Он отбрасывает отрицание равенства в дизъюнкции $(\neg(x = t) \& A(x) \vee B_1(x) \vee \ldots \vee B_n(x))$ после проверки того, что $A(t)$ вытекает из некоторого $B_i(t)$. Уровень срабатывания приема равен 4.

2. Переобозначение связанных переменных для отождествления кванторов. Если в преобразуемом терме обнаруживаются два отличающихся друг от друга квантора существования, имеющих одинаковую длину, то предпринимается попытка переобозначить связанные переменные так, чтобы кванторы отождествились. Уровень срабатывания приема равен 5. Он играет вспомогательную роль, подготавливая возможность последующих преобразований дизъюнкции.

3. Исключение модуля. Если преобразуемый терм содержит подвыражение вида $\{F(|A|), F(-|A|)\}$, то оно заменяется на $\{F(A), F(-A)\}$. Причина, по которой такой прием пришлось включить в нормализатор "нормили", состоит в том, что условие принадлежности неизвестной конечному списку обычно возникает при его работе после обработки дизъюнкции равенств. Дальнейшее упрощение списка за счет приемов сканирования задачи уже невозможно - ответ будет выдан сразу по окончании применения нормализатора. Поэтому приходится включать в него множество мелких приемов упрощения ответа, срабатывание которых становится достаточно вероятным из-за предшествующей деятельности данного нормализатора.

1.7 Приемы символа "не"

Эти приемы, в основном, крайне просты и обеспечивают лишь общую стандартизацию утверждений.

Двойное отрицание

Утверждение "не(не(A))" заменяется на A. Уровень срабатывания равен 0.

Отрицание дизъюнкции

Утверждение "не(или(A_1 … A_n))" заменяется на "и(не(A_1) … не(A_n))". Уровень срабатывания - 0.

Отрицание квантора общности

Утверждение "не(длялюбого($x_1 \ldots x_n$ если $A_1 \ldots A_n$ то A_0))" заменяется на "существует($x_1 \ldots x_n$ и($A_1 \ldots A_n$ не(A_0)))". Уровень срабатывания - 0.

Отрицание квантора существования

Утверждение "не(существует($x_1 \ldots x_n$ и($A_1 \ldots A_n$)))" заменяется на "длялюбого(x_1 … x_n если $A_1 \ldots A_{n-1}$ то не(A_n))". Уровень срабатывания равен 1. Заметим, что в этом преобразовании можно произвольно перегруппировывать утверждения между антецедентами и консеквентом. Прием относит к антецедентам первые конъюнктивные члены. Вообще говоря, это не всегда целесообразно. От способа группировки может зависеть, например, вывод следствий, основанный на использовании кванторной импликации, обратный вывод, и т.п. Поэтому имеются специальные приемы, обеспечивающие перегруппировку утверждений кванторной импликации между антецедентами и консеквентом в соответствии с требованиями контекста.

Отрицание эквивалентности

Если кванторная импликация имеет антецедент вида "не(эквивалентно(A B))", причем консеквент ее не имеет заголовка "эквивалентно", то происходит контрапозиция консеквента с данным антецедентом. В прочих случаях утверждение "не(эквивалентно(A B))" преобразуется к виду "эквивалентно(A не(B))". Уровень срабатывания - 1.

Отрицание конъюнкции

Если преобразовать отрицание конъюнкции "не($A_1\&\dots\&A_n$)" в дизъюнкцию отрицаний "не(A_1) $\vee\dots\vee$ не(A_n)", то может оказаться, что утеряно сопровождение по о.д.з. для некоторых из дизъюнктивных членов "не(A_i)". Именно, это произойдет, если данное сопровождение обеспечивалось другими A_j. Поэтому преобразование применяется с некоторой модификацией. Сначала для каждого A_i проверяется, что все сопровождающие его по о.д.з. утверждения B_i, не вытекающие из контекста отрицания конъюнкции, суть следствия прочих A_j. Тогда при переходе к дизъюнкции отрицание A_i берется в конъюнкции с утверждениями B_i. Эквивалентность такого перехода основана на отсутствии циклов в отношении сопровождения по о.д.з. между различными утверждениями. Если же указанное выше условие не выполняется, то преобразование блокируется, причем вводится комментарий, предотвращающий повторные попытки его применения. Уровень срабатывания приема равен 2. В качестве примера можно привести преобразование утверждения $\neg(x/y = z\ \&\ \neg(y = 0))$. Второй конъюнктивный член является сопровождающим по о.д.з. для первого. Поэтому преобразованное утверждение будет иметь вид $(\neg(x/y = z)\ \&\ \neg(y = 0)) \vee (y = 0)$.

Отрицание условия либо посылки

Если для условия "не(A)" усматривается, что A - посылка либо другое условие, то первое условие заменяется на константу "ложь". Если "не(A)" - посылка, то проверяется наличие условия A, которое заменяется на "ложь". Уровень срабатывания обоих приемов равен 1.

Контрапозиция для условия задачи на доказательство

Если условие задачи на доказательство имеет вид "не(A)", а некоторая посылка - вид "не(B)", то условие заменяется на B, а посылка - на A. Для проверки целесообразности такой контрапозиции применяется справочник "не", обращение к которому происходит на символе, являющемся заголовком утверждения A.

1.8 Приемы символа "длялюбого"

Частота возникновения кванторных конструкций в задачах сильно зависит от рассматриваемой предметной области. Например, в геометрии они встречаются редко; в теоретических задачах по математическому анализу - часто. Типичный способ исключения кванторных конструкций - ввод определений, с помощью которых они сводятся к новым бескванторным утверждениям или выражениям. Так, для кванторной импликации $\forall_x(x \in A \to x \in B)$ вводится по определению обозначение $A \subseteq B$; для выражения $\mathrm{set}_x(\exists_y(y \in A\ \&\ x = f(y)))$ - обозначение "образ(f,A)", и т.д. За счет наработки теорем, обслуживающих введенные новые понятия, во многих разделах удается избежать необходимости расшифровки по определениям, и сделать процессы решения задач почти "бескванторными". Однако, остается достаточное количество случаев, когда исключение кванторных конструкций невозможно, и для работы с ними приходится создавать множество специальных приемов. На долю квантора общности приходится наибольшее число таких приемов.

Простейшая стандартизация

Сначала перечислим приемы, срабатывающие на нулевом уровне.

1. Переобозначение связанных переменных. Если в задаче встречается квантор или описатель, то связанные переменные его немедленно переобозначаются так, чтобы они не имели свободных вхождений в утверждения, образующие внешний контекст для данного квантора или описателя. Например, если внешний контекст кванторного утверждения $\forall_x P(x)$ содержит равенство $x = 2$, то это утверждение будет заменено на $\forall_y P(y)$. Такое преобразование выполняется оператором "нормализациясвязок". Возникающая после переобозначений стандартизация гарантирует, что на уровнях, больших или равных 1, никакая связанная переменная не будет совпадать с обозначением объекта, уже имеющимся в текущем контексте.

2. Устранение из кванторной приставки дублирующих переменных. Если кванторная приставка имеет переменные, встречающиеся более одного раза, то повторные вхождения переменных из нее исключаются.

3. Устранение дублирующих антецедентов. Если имеются повторяющиеся антецеденты, то они исключаются.

4. Исключение из кванторной приставки переменных, не имеющих свободных вхождений в антецеденты и консеквент. Если некоторые переменные связывающей приставки избыточны - не имеют свободных вхождений ни в антецеденты, ни в консеквент, то они удаляются. В случае, когда связывающая приставка остаетя пустой, происходит замена кванторной импликации $\forall_{x_1 \ldots x_n}(A_1 \,\&\ldots\&\, A_n \to A_0)$ на дизъюнкцию $\neg A_1 \vee \ldots \vee \neg A_n \vee A_0$.

5. Устранение дизъюнкции в консеквенте. Если кванторная импликация имеет вид $\forall_{x_1 \ldots x_n}(A_1 \,\&\ldots\&\, A_n \to (B_1 \vee \ldots \vee B_m))$, то она преобразуется к виду $\forall_{x_1 \ldots x_n}(A_1 \,\&\ldots\&\, A_n \,\&\, \neg B_1 \,\&\ldots\,\&\, \neg B_{m-1} \to B_m)$. Исключение составляет случай редактирования теоремы, специально предназначенной для указания списка альтернатив.

6. Замена конъюнктивного антецедента на группу антецедентов. Если некоторый антецедент представляет собой конъюнкцию утверждений, то он исключается, а все его конъюнктивные члены становятся новыми антецедентами.

7. Усмотрение истинности кванторной импликации, у которой антецедент совпадает с консеквентом либо имеются два противоположных антецедента. В указанной ситуации кванторная импликация заменяется на константу "истина".

8. Исключение антецедента, отрицание которого совпадает с консеквентом.

9. Усмотрение антецедентов, указывающих несовместимые типы одной и той же переменной. Если кванторная импликация имеет антецеденты $P(x)$ и $Q(x)$, где P, Q - названия различных типов объектов, причем с помощью справочника "род" усматривается, что ни один из типов не является подтипом другого, то эта импликация заменяется на константу "истина".

Далее идут приемы стандартизации кванторных импликаций, уровень срабатывания которых больше нуля.

1. Устранение конъюнкции в консеквенте. Если кванторная импликация имеет вид $\forall_{x_1 \ldots x_n}(A_1 \,\&\ldots\&\, A_n \to (B_1 \,\&\ldots\&\, B_m))$, то ее можно преобразовать к виду конъюнкции кванторных импликаций
$$\forall_{x_1 \ldots x_n}(A_1 \,\&\ldots\&\, A_n \to B_1) \,\&\ldots\&\, \forall_{x_1 \ldots x_n}(A_1 \,\&\ldots\&\, A_n \to B_m).$$
Это преобразование, вообще говоря, полезно, так как упрощает кванторные конструкции, хотя и увеличивает их число. Однако, есть ряд ограничений на его применение. Во-первых, нецелесообразно применять его к условию задачи на доказательство, так как выгоднее сначала применить другое преобразование, исключающее кванторную импликацию за счет переброски антецедентов в список посылок (см. ниже). Во-вторых, при преобразовании условия задачи на описание нужно контролировать сохранение сопровождения по о.д.з. Если какое-то B_i обеспечивало о.д.з. для другого B_j, то их невыгодно разбивать по разным кванторным импликациям. Поэтому в данном случае преобразование модифицируется, аналогично преобразованию отрицания конъюнкции. Находятся все B_i, не используемые для сопровождения по о.д.з. других B_j, и утверждения $B_1, \ldots, B_m$ распределяются по группам, образованным указанными B_i и сопровождающими их по о.д.з. другими B_k. Эти группы могут пересекаться. Затем исходная импликация заменяется на конъюнкцию импликаций, соответствующих найденным группам. Уровень срабатывания приема равен 1.

2. Устранение дизъюнкции в антецеденте. Если кванторная импликация имеет вид $\forall_{x_1 \ldots x_n}(A_1 \,\&\ldots\&\, A_{i-1} \,\&(B_1 \vee\ldots\vee B_m) \,\&\, A_{i+1} \,\&\ldots\&\, A_n \to A_0)$, то она заменяется на конъюнкцию импликаций $\forall_{x_1 \ldots x_n}(A_1 \,\&\ldots\&\, A_{i-1} \,\&\, B_1 \,\&\, A_{i+1} \,\&\ldots\&\, A_n \to A_0) \,\&\ldots\&\, \forall_{x_1 \ldots x_n}(A_1 \,\&\ldots\&\, A_{i-1} \,\&\, B_m \,\&\, A_{i+1} \,\&\, \ldots\&\, A_n \to A_0)$. Прием применяется без ограничений, и уровень срабатывания его равен 1.

3. Усмотрение вхождения антецедента в консеквент либо в другой антецедент.

   Если некоторый антецедент кванторной импликации входит в другой ее антецедент либо в консеквент, то это вхождение заменяется на константу "истина". Если антецедент имеет вид "не(A)", причем утверждение A входит в другой антецедент либо в консеквент, то вхождение этого утверждения заменяется на константу "ложь". Уровень срабатывания равен 1.

4. Усмотрение вхождения консеквента в антецедент. Если консеквент кванторной импликации входит в ее антецедент, то это вхождение заменяется на константу "ложь". Если консеквент имеет вид "не(A)", причем утверждение A входит в антецедент, то его вхождение заменяется на константу "истина". Уровень срабатывания равен 1.

5. Сдвоенные переменные кванторной приставки. Если кванторная импликация $\forall_{x_1 \ldots x_k}(A_1 \,\&\ldots\&\, A_n \to A_0)$ имеет такие две различные переменные x_i, x_j, которые повсюду встречаются вместе под одной и той же операцией f, кроме, быть может, антецедентов, указывающих тип значения этих переменных по отдельности, то предпринимается попытка перейти от пары переменных x_i, x_j к новой переменной, обозначающей $f(x_i, x_j)$. Для коммутативной и ассоциативной операции f проверяется, что как x_i, так и x_j присутствуют среди ее операндов лишь однократно. Для не коммутативной операции f проверяется, что она двуместная, причем операнды x_i, x_j встречаются везде в одном и том же порядке. Чтобы определить условия на допустимые значения новой переменной

z, обозначающей $f(x_i, x_j)$, решается вспомогательная задача, условия которой суть $z = f(x_i x_j)$, $P(z)$, где P - тип значений операции f, а также все антецеденты, имеющие единственную переменную x_i либо единственную переменную x_j. Неизвестные задачи суть переменные x_i, x_j. На ответ навешивается квантор существования по x_i, x_j, и результат упрощается. Найденные условия на z берутся в качестве антецедентов, заменяющих ранее имевшиеся антецеденты, содержащие x_i, x_j по отдельности. Все сдвоенные вхождения x_i, x_j в антецеденты и консеквент заменяются на z; в связывающей приставке проводится такая же замена. Фактически прием даже не переходит к новой переменной z, а использует в качестве нее старую переменную x_i. Уровень срабатывания равен 2.

6. Лексикографическое упорядочение антецедентов. По завершении описанных выше преобразований общей стандартизации кванторной импликации предпринимается лексикографическое переупорядочение ее антецедентов. Уровень срабатывания приема равен 3. Такое переупорядочение может быть необходимым для усмотрения совпадения двух различных импликаций.

Декомпозиция кванторной импликации на независимые утверждения

Утверждения $A_1, \ldots, A_n, \neg A_0$ кванторной импликации $\forall_{x_1 \ldots x_k}(A_1 \,\&\ldots\&\, A_n \to A_0)$ могут распадаться на такие подмножества $B_1, \ldots, B_r$, что элементы различных подмножеств не зависят от общих переменных кванторной приставки $x_1, \ldots, x_k$. В этом случае становится возможной декомпозиция кванторной импликации в дизъюнкцию некоторых утверждений $C_1, \ldots, C_r$. Число r выбирается наибольшим возможным, $r > 1$, что соответствует выбору в качестве каждого B_i одной компоненты связности исходного списка утверждений для отношения зависимости двух утверждений от общей переменной кванторной приставки. C_i есть кванторная импликация, полученная из элементов подмножества B_i выбором в качестве консеквента отрицания одного из них, а в качестве антецедентов - оставшихся элементов. Кванторная приставка у C_i представляет собой подмножество K_i переменных кванторной приставки исходной импликации, встречающихся в B_i. Если K_i пустое, то B_i, очевидно, одноэлементно, и тогда C_i является отрицанием этого элемента. Прием применяется на уровнях 0 либо 2. Уровень 0 имеет место для задач на описание, имеющих цель "кванторная-свертка". Здесь необходимо успеть выполнить декомпозицию до того, как кванторная импликация будет преобразована "по определениям" в бескванторное утверждение. При наличии в кванторной импликации логических констант прием блокируется, так как целесообразно сначала от них избавиться.

Для получения подмножеств B_i используется процедура "компонента(x1 x2 x3)", при обращении к которой задаются набор термов x2 и два списка переменных - x1 и x3. x1 указывает свободные переменные некоторого терма T набора x2. Процедура находит содержащую T компоненту связности в x2, состоящую их всех термов, к которым можно перейти от T по цепочке переходов между термами, зависящими от общей переменной списка x3.

Выбор консеквента в кванторной импликации C_i, вообще говоря, неоднозначен. Прием делает некоторый предварительный выбор, руководствуясь рядом эвристических соображений. В частности, он пытается сохранить старый консеквент, если таковой попал в C_i, и сохранить в антецедентах простейшие утверждения, обеспечивающие о.д.з. для более сложных утверждений. При неудачном выборе консеквента необходимые перестановки будут выполнены другими приемами.

Контрапозиции

Приведем несколько общих приемов, переставляющих местами консеквент и антецедент кванторной импликации (разумеется, переходя при этом к их отрицаниям). Такие преобразования называются контрапозициями.

1. Консеквент и антецедент начинаются с символа "не". В этом случае обычно выполняется контрапозиция, позволяющая отбросить оба отрицания. Однако, в ряде специальных случаев применение приема блокируется. Например, если консеквент имеет вид отрицания равенства, содержащего неизвестные, а антецедент неизвестных не содержит, либо если антецедент необходим для обеспечения сопровождения по о.д.з. Уровень срабатывания приема равен 0.

2. Консеквент имеет вид "не($P(x)$)", где x - переменная кванторной приставки, а P - логический символ. Если некоторый антецедент не имеет вида $Q(y)$, где y - переменная кванторной приставки, а Q - логический символ, то для него выполняется контрапозиция. Уровень срабатывания равен 1.

3. Консеквент имеет вид "не($P(x)$)", и никакой антецедент не имеет своим заголовком символа "не". Составляется список S антецедентов, пополненный утверждением $P(x)$. Находится подмножество R утверждений списка S, не используемых в качестве сопровождающих по о.д.з. для других утверждений данного списка. Если R состоит из единственного элемента Q, отличного от $P(x)$, то выполняется контрапозиция для антецедента Q. Если R более чем одноэлементно, то контрапозиция выполняется для такого его элемента Q, который либо является кванторной импликацией, в то время как $P(x)$ элементарно, либо имеет наибольшее число входящих в него переменных кванторной приставки, превосходящее такое число для $P(x)$.

4. Некоторый антецедент имеет вид "не(P)", а консеквент Q имеет тот же заголовок, что и утверждение P. Если список свободных переменных консеквента включается в список свободных переменных утверждения P, длина терма Q хотя бы в 1.2 раза меньше длины терма P, и антецедент "не(P)" не используется для сопровождения по о.д.з., то выполняется контрапозиция.

Слияние кванторных приставок

Если кванторная импликация имеет вид "длялюбого($x_1 \dots x_n$ длялюбого($y_1 \dots y_m$ если $A_1 \dots A_k$ то A_0))", то она преобразуется к виду "длялюбого($x_1 \dots x_n y_1 \dots y_m$ если $A_1 \dots A_k$ то A_0)". Уровень срабатывания равен 0.

Группировка кванторов наружу

На уровне 1 выполняется обобщающая предыдущий случай группировка кванторов наружу. Именно, если кванторная импликация имеет вид "длялюбого($x_1 \dots x_n$ если $B_1 \dots B_p$ то длялюбого($y_1 \dots y_m$ если $A_1 \dots A_k$ то A_0))", то она заменяется на "длялюбого($x_1 \dots x_n y_1 \dots y_m$ если $B_1 \dots B_p A_1 \dots A_k$ то A_0)". Аналогично, утверждение "длялюбого($x_1 \dots x_n$ если $A_1 \dots A_{i-1}$ существует($y_1 \dots y_m$ B) $A_{i+1} \dots A_k$ то A_0)" заменяется на "длялюбого($x_1 \dots x_n y_1 \dots y_m$ если $A_1 \dots A_{i-1} B A_{i+1} \dots A_k$ то A_0)". В последнем случае требуется, чтобы рассматриваемый терм задачи не содержал выражения "отображение($y_1 \dots y_m$ B t)".

Исключение избыточных антецедентов

Несколько простых приемов позволяют отбрасывать антецеденты, не изменяющие области истинности кванторной импликации. Обычно такие приемы бывают нужны при выводе теорем.

1. Отбрасывается антецедент $P(x)$, где одноместный предикатный символ P определяет тип объекта, а связанная переменная x не встречается в других антецедентах и в консеквенте. Уровень срабатывания равен 0.

2. Если некоторая переменная кванторной приставки x не встречается в консеквенте, то рассматриваются все содержащие ее антецеденты $A_1, \ldots, A_n$. Если из оставшихся антецедентов $B_1, \ldots, B_m$ вытекает существование такого значения x, что все утверждения $A_1, \ldots, A_n$ истинны, то последние отбрасываются из списка антецедентов. Для проверки здесь используется вспомогательная задача на описание, решаемая с сильным ограничением на допустимую трудоемкость. Прием применяется только при выводе теорем - для его активизации требуется цель "редуцирование". Уровень срабатывания равен 5.

3. Если антецедент кванторной импликации, возникающей при выводе теорем, имеет вид $\neg(x = t)$, где x - переменная кванторной приставки, не входящая в t, то проверяется, что теорема сохраняет истинность и в случае $x = t$. После этого антецедент отбрасывается. Уровень срабатывания равен 5.

Замена переменной кванторной приставки

Пусть кванторная импликация "длялюбого($x_1 \ldots x_n$ если $A_1 \ldots A_m$ то A_0)" имеет переменную x_i своей кванторной приставки, которая встречается в консеквенте только внутри вхождений одного и того же выражения T, причем все содержащие x_i антецеденты не имеют переменных, отличных от x_i. В качестве T берется минимальное выражение, корневым операндом которого служит x_i, причем проверяется, что число вхождений T в консеквент не менее двух и что ни одна из свободных переменных T не связана никаким внешним квантором или описателем, находящимся в консеквенте. В указанной ситуации предпринимается попытка заменить переменную x_i на новую переменную y, являющуюся вспомогательным обозначением для T. Чтобы найти условия на область изменения новой переменной, преобразуется к бескванторному виду утверждение "существует(x_i и(Q $y = T$))", где Q - конъюнкция содержащих x_i антецедентов. Преобразование выполняется в два этапа - сначала решается задача на описание с условиями Q, $y = T$ и несущественной неизвестной x_i, затем упрощается квантор существования. Если результат упрощения имеет вид $P_1 \vee \ldots \vee P_k$, $k \geq 1$, то проверяется, что каждый содержащий y конъюнктивный член каждого P_j не содержит других переменных. Далее вводятся кванторные импликации C_j, $j = 1, \ldots, k$, полученные из исходной кванторной импликации заменой в консеквенте всех вхождений выражения T на y, заменой всех содержащих x_i антецедентов на P_j и заменой в кванторной приставке x_i на y. Наконец, предпринимается замена исходной кванторной импликации на конъюнкцию результатов упрощения кванторных импликаций $C_1, \ldots, C_k$. Уровень срабатывания равен 4.

Переформулировка кванторной импликации как условия пустоты класса

Если в одном и том же терме задачи присутствуют кванторная импликация "длялюбого($x_1 \ldots x_n$ если $A_1 \ldots A_m$ то A_0)" и выражение "отображение($x_1 \ldots x_n$ P t)", причем отрицание кванторной импликации совпадает, с точностью до простейших преобразований, с утверждением "существует($x_1 \ldots x_n$ P)", то кванторная импликация преобразуется к виду "равно(класс($x_1 \ldots x_n$ P)пусто)". Уровень срабатывания равен 4.

Импликативное условие задачи на доказательство

Если задача на доказательство Z имеет условие вида "длялюбого($x_1 \ldots x_n$ если A_1 $\ldots A_m$ то A_0)", то она сводится к новой задаче на доказательство Z', получаемой из текущей добавлением утверждений $A_1, \ldots, A_m$ к списку посылок и выбором утверждения A_0 в качестве условия. Комментарии к посылкам и комментарии к задаче переносятся из Z в Z' с помощью справочников "новаяпосылка", "длялюбого". Задача Z' решается с тем же максимальным уровнем, что и Z. Результат ее решения (ответ "истина" либо символ "отказ") выдается как ответ на задачу Z. Уровень срабатывания равен 3.

Контрапозиция кванторной импликации в посылках задачи на доказательство и ее условия, имеющего вид отрицания

Если задача на доказательство Z имеет единственную кванторную импликацию A в посылках, а условие ее имеет вид "не(B)", то она сводится к новой задаче Z', полученной заменой посылки A на B, а условия - на отрицание A. Целью такой контрапозиции является получение в условии задачи квантора существования для последующего перехода от задачи на доказательство к задаче на подбор примера. Уровень срабатывания приема равен 6.

Обратный вывод в задаче на доказательство

Если задача на доказательство имеет своей посылкой кванторную импликацию "длялюбого($x_1 \ldots x_n$ если $A_1 \ldots A_m$ то A_0)", у которой все переменные связывающей приставки входят в консеквент A_0, а условием задачи служит элементарное утверждение B с тем же заголовком, что A_0, отличным от равенства, то проверяется, не является ли B результатом подстановки в A_0 некоторых термов $t_1, \ldots, t_n$ вместо переменных $x_1, \ldots, x_n$. Если такие термы найдены, то последовательно рассматриваются результаты B_i подстановки их вместо переменных кванторной приставки в антецеденты $A_1, \ldots, A_m$. Для каждого B_i предпринимается попытка усмотреть его истинность в контексте посылок текущей задачи, используя вспомогательную задачу на доказательство с замороженными посылками и с достаточно сильным ограничением на трудоемкость. Таким образом, реализуется цикл попыток быстрого усмотрения. В случае его успешного завершения на текущую задачу выдается ответ "истина". Уровень срабатывания приема равен 5.

Использование кванторной импликации, имеющей заголовком консеквента квантор существования, для доказательства существования

Если задача на доказательство имеет посылку вида "длялюбого($x_1 \ldots x_n$ если $A_1 \ldots$ A_m то существует($y_1 \ldots y_k$ и($B_1 \ldots B_r$)))", а условие ее имеет вид "существует($z_1 \ldots z_p$ и($C_1 \ldots C_q$))", то предпринимается попытка подобрать такую подстановку термов $s_1, \ldots, s_p$ вместо переменных $z_1, \ldots, z_p$ в утверждения $C_1, \ldots, C_q$, которая дает подмножество результатов подстановки некоторых термов $t_1, \ldots, t_n$ вместо переменных

$x_1, \ldots, x_n$ в утверждения $B_1, \ldots, B_r$. Для этого используется процедура "унификация". В результате оказываются найдены конкретные значения переменных квантора существования - условия задачи, отождествляющие подкванторные утверждения с некоторыми конъюнктивными членами консеквента кванторной импликации. Далее остается только проверить истинность утверждений, получающихся при подстановке в антецеденты термов $t_1, \ldots, t_n$ вместо переменных $x_1, \ldots, x_n$. Это делается с помощью оператора "извлекается", решающего вспомогательную задачу на доказательство с ограниченными средствами, обеспечивающими попытку быстрого усмотрения. Уровень срабатывания данного приема равен 1. Увеличение его привело бы к тому, что прием оказался практически отключенным: на уровне 2 срабатывает прием, сводящий доказательство существования к задаче на подбор примера.

Разрешение подкванторных утверждений относительно переменных кванторной приставки

В некоторых случаях подкванторные утверждения допускают явное разрешение относительно переменных кванторной приставки как неизвестных. Обычно, вслед за этим удается вообще избавиться от данного квантора, переформулировав его в терминах элементарных утверждений. Для удобства кванторная импликация преобразуется к виду отрицания квантора существования, и рассматриваются подкванторные утверждения последнего. Например, если условие задачи на описание имеет вид $\forall_x(1 \leq x \;\&\; x \leq 3 \to x + a < 2)$, то можно решить неравенства $1 \leq x$, $x \leq 3$, $\neg(x + a < 2)$ относительно несущественной неизвестной x. В процессе решения будут исключаться явные указания интервалов допустимых значений x и сохраняться только условия на параметры, при которых эти значения существуют. Получив ответ $a \geq -1$, преобразуем исходную кванторную импликацию к виду $\neg\exists_x(a \geq -1)$, что далее дает бескванторное условие $a < -1$.

Приведенный прием часто бывает полезен в задачах по элементарной алгебре; в других разделах он обычно мало эффективен и приводит лишь к неоправданным дополнительным затратам времени. Поэтому начинается прием с проверки серии эвристических фильтров, ограничивающих его активизацию.

Прием используется только для кванторных импликаций "длялюбого($x_1 \ldots x_n$ если $A_1 \ldots A_m$ то A_0)", являющихся условиями задач на описание. Проверяется, что такая кванторная импликация не является частью ответа, задающей серию ограничений. Находится список утверждений $A_1, \ldots, A_m, \neg A_0$, и решается вспомогательная задача на описание с данным списком условий. Посылками ее служат все утверждения из контекста рассматриваемой кванторной импликации; несущестенными неизвестными - переменные $x_1, \ldots, x_n$. Если найден ответ B, то кванторная импликация заменяется на утверждение $\neg\exists_{x_1 \ldots x_n} B$. Уровень срабатывания приема равен 3.

Фиксация значения связанной переменной через равенство в антецеденте

Кванторная импликация вида "длялюбого($x_1 \ldots x_n$ если $A_1 \ldots\; A_{i-1}\; x_j = t\; A_{i+1} \ldots\; A_m$ то A_0)", где выражение t не содержит x_j, за исключением крайне редких специальных случаев, преобразуется к виду "длялюбого($x_1 \ldots x_{j-1} x_{j+1} \ldots x_n$ если $B_1 \ldots B_{i-1} B_{i+1} \ldots B_m$ то B_0)". Здесь каждое B_k - результат подстановки выражения t вместо переменной x_j в утверждение A_k. Чтобы преобразование было корректным, связанные переменные утверждения A_k переобозначаются на переменные, отличные от свободных переменных терма t. Уровень срабатывания приема равен 0.

Фиксация значения связанной переменной через отрицание равенства в консеквенте

Этот прием аналогичен предыдущему. Он заменяет кванторную импликацию вида "длялюбого($x_1 \dots x_n$ если $A_1 \dots\ A_m$ то не($x_j = t$))" на импликацию "длялюбого($x_1 \dots x_{j-1} x_{j+1} \dots x_n$ если $B_1 \dots B_{m-1}$ то не(B_m))". Уровень срабатывания его тоже равен 0.

Вывод следствий с помощью кванторной импликации

Для вывода следствий в посылках некоторой задачи можно использовать кванторную импликацию "длялюбого($x_1 \dots x_n$ если $A_1 \dots A_m$ то A_0)", содержащуюся в тех же посылках. Имеется несколько приемов, осуществляющих такой вывод:

1. Вывод без контрапозиций. В случае задачи на доказательство уровень срабатывания приема равен 2, иначе он равен 3. Проверяется, что все утверждения $A_1, \dots, A_m, A_0$ элементарны (не содержат логических сязок, кроме, быть может, внешнего отрицания, и не имеют связанных переменных). Проверяется также отсутствие некоторых специальных комментариев, делающих использование данной кванторной импликаций для прямого вывода следствий нецелесообразным. После этого предпринимается обращение к процедуре "подборзначений", которая пытается подобрать такие термы $t_1, \dots, t_n$, подстановка которых в антецеденты дает (с точностью до простейших преобразований) некоторые посылки $B_1, \dots, B_n$. Так как целью вывода является получение утверждений, связывающих между собой переменные задачи, то проверяется, что указанные посылки имеют хотя бы одну свободную переменную. Проверяется, что каждая посылка B_i либо имеет свободную переменную, встречающуюся в условии задачи, либо относится к той же компоненте списка посылок, определяемой отношением зависимости двух утверждений от общей переменной, что и рассматриваемая кванторная импликация. После этого определяется результат C подстановки термов $t_1, \dots, t_n$ вместо переменных $x_1, \dots, x_n$ в консеквент A_0. Предпринимается упрощение утверждения C, в котором участвуют нормализаторы общей стандартизации и (в случае равенства) оператор "стандравно". Те подтермы данного утверждения, которые уже встречаются в посылках задачи, упрощением не затрагиваются. Если в результате упрощения получено утверждение D, отличное от константы "истина", то выполняется обращение к оператору "фильтримпликации" для окончательного определения целесообразности вывода следствия D.

   Здесь мы сталкиваемся с явлением, характерным для многих общих приемов. Ввиду их общности, иногда бывает затруднительно дать какие-то простые рекомендации относительно того, в каких случаях прием применять нужно, а в каких - не нужно. Чтобы обеспечить хорошее управление приемом, приходится вводить в него накопитель эвристических решающих правил, имеющих существенно меньший уровень общности, чем логическая схема выполняемых преобразований. Эти правила могут быть связаны с понятиями различных предметных областей, и тогда для объединения их в общую процедуру применяются справочники. После того, как ситуация конкретизируется, обычно удается найти достаточно хорошие мотивировки для разрешения либо блокировки срабатывания. В нашем случае несколько различных решающих правил, накопленных при обучении системы, сгруппированы в процедуре "фильтримпликации".

Сразу заметим, что для тех разделов, по которым велось обучение решателя, кванторные импликации возникали в посылках исключительно редко. Поэтому обучающий материал для создания фильтров данного приема оказался небольшим. Это привело к тому, что длительное время достаточными являлись самые простые средства отсечения. Разрешалось выводить следствия, у которых новые подвыражения не превосходили по своей длине подвыражений, уже имеющихся в задаче. Лишь при переходе к рассмотрению простых "теоретических" задач по математическому анализу данный прием стал порождать чрезмерное количество бесполезных следствий, и процедура "фильтримпликации" пополнилась дополнительными ограничителями, относящимися к вводу функциональных переменных (т.е. термов вида "значение($f\ t$)"). Эти ограничители достаточно грубые и впоследствии должны быть существенно уточнены.

Перейдем к перечислению решающих правил процедуры "фильтримпликации":

(a) Если утверждение D содержит подвыражение "значение($f\ t$)", причем текущая задача на описание имеет цель "пример", и некоторое ее условие также содержит подвыражение вида "значение($f\ s$)", не связанное внешним квантором либо описателем, а посылки такого вида нет, то принимается решение о выводе следствия D.

(b) Если предыдущая ситуация не имеет места, причем D содержит подвыражение "значение($f\ t$)", где t имеет вхождение символа "значение", то вывод блокируется. Хотя и несколько искусственное, это ограничение должно восприниматься в контексте наличия других приемов, специально предназначенных для вывода следствий с подвыражениями типа "значение($\ldots$)". Поэтому для данного приема оставлены лишь простейшие ситуации вывода следствий, содержащих такие подвыражения.

(c) Если D содержит подвыражение "значение($f\ t$)", которое до этого не встречалось в задаче, то проверяется наличие такой функциональной переменной g, что выражения "значение($f\ \ldots$)" входят в отличные от рассматриваемой кванторной импликации термы задачи только внутри выражений "значение($g\ \ldots$)", причем хотя бы одно такое вхождение имеется. В этой ситуации вывод следствия D блокируется.

(d) Если D содержит подвыражение "значение($\ldots$)", не встречающееся в условиях и посылках текущей задачи на описание, имеющей цель "пример", причем условие из пункта а) не выполнено, то вывод блокируется.

(e) Если решается задача на описание, имеющая цель "пример", либо задача на доказательство, максимальный уровень которой более 6, причем длина каждого подвыражения утверждения D не превосходит длины хотя бы какого-то подвыражения посылки либо условия задачи, то принимается решение о выводе следствия D.

(f) Если каждое подвыражение утверждения D уже имеется в посылках и условиях задачи, то принимается решение о выводе следствия D.

(g) Если кванторная импликация снабжена комментарием "то", блокирующим ограничения процедуры "фильтримпликации", то принимается решение о выводе следствия D. Однако, после получения с помощью рассматриваемой кванторной импликации более 4 следствий, комментарий "то" к ней удаляется.

(h) Если решается задача на исследование, имеющая цель"известно", причем D содержит подвыражение специального вида ("значение(...)" либо числовой атом), содержащее неизвестные и уже имеющееся в некоторой посылке задачи, в которую входят также неизвестные внешней задачи на описание, то принимается решение о выводе следствия D.

Если процедура "фильтримпликации" принимает решение о выводе следствия D, то оно регистрируется в списке посылок. При этом веса всех посылок и условий, имеющих общую свободную переменную с D, уменьшаются до 1.

2. Вывод с контрапозициями.

   Уровень срабатывания приема равен 5. Здесь перед использованием кванторной импликации для вывода следствий разрешается выполнить контрапозицию какого-либо ее антецедента (поменять его местами с консеквентом, заменив оба утверждения на их отрицания). Технически это достигается путем составления списка $A_1, \ldots, A_m, \neg A_0$ и обращением к процедуре "подборпосылок", которая пытается идентифицировать с посылками задачи все утверждения списка, кроме какого-либо одного. На основе отрицания последнего и строится следствие D. Это делается аналогично предыдущему приему. Фильтрация выводимых утверждений D усиливается - пропускаются лишь те утверждения, все подвыражения которых уже встречались в задаче. Соответственно, обращение к процедуре "фильтримпликации" для данного приема становится не нужным. Уменьшение весов посылок и условий не предпринимается.

3. Вывод для получения утверждения с функциональным выражением, уже имеющимся в задаче. Если решается задача на исследование, посылка которой содержит функциональное выражение "значение(f t)", зависящее от неизвестных внешней задачи на описание, то могут предприниматься попытки вывода следствий, содержащих данное функциональное выражение. Прием, выполняющий такие попытки, инициируется при обнаружении в консеквенте A_0 кванторной импликации подвыражения "значение(f s)", где f - переменная. Он проверяет, что все переменные кванторной приставки содержатся в выражении s, выбирает в некоторой элементарной посылке текущей задачи на исследование подвыражение "значение(f t)", где t зависит от неизвестных внешней задачи на описание, и находит такие термы $r_1, \ldots, r_n$, подстановка которых вместо переменных $x_1, \ldots, x_n$ в выражение s дает выражение t. Находятся результаты $B_1, \ldots B_m, B_0$ подстановки термов $r_1, \ldots, r_n$ вместо переменных $x_1, \ldots, x_n$ в антецеденты и консеквент кванторной импликации. Используя проверочные операторы, прием пытается установить истинность утверждений $B_1, \ldots B_n$ в контексте списка посылок текущей задачи. Здесь устанавливается достаточно сильное ограничение на допустимую трудоемкость. В случае успеха следствие B_0 регистрируется в посылках задачи. Уровень срабатывания приема равен 4.

4. Вывод равенства для числового атома. Если решается задача на исследование, и консеквент кванторной импликации представляет собой тождество $F(a_1 \ldots a_k) = t$, где F - некоторая числовая характеристика нечисловых объектов (длина, масса, и т.п.), то предпринимается попытка установить с его помощью значения данной характеристики, встречающиеся в прочих посылках задачи. Для любого вхождения в такую посылку подтерма $F(b_1 \ldots b_k)$, не связанного внешними кванторами и описателями, находится подстановка p вместо переменных

кванторной приставки, переводящая $a_1, \ldots, a_k$ в $b_1, \ldots, b_k$. Проверяется, что истинность результатов применения p к антецедентам усматривается из посылок задачи с помощью проверочных операторов, и регистрируется следствие $F(b_1 \ldots b_k) = s$, где s - результат применения p к терму t. Уровень срабатывания приема равен 4.

5. Попытка вывести следствие, содержащее уже встречавшееся в задаче выражение с неизвестными. Если решается задача на исследование, причем длина кванторной приставки больше 1, то рассматривается некоторое более чем однобуквенное подвыражение t ее консеквента, заголовок которого не ассоциативен и отличен от символа "значение". Это выражение берется максимальным в том смысле, что является непосредственным операндом некоторого утверждения. Проверяется, что пересечение S свободных переменных выражения t с кванторной приставкой непусто и что непуст список R его свободных переменных, не входящих в кванторную приставку. Затем ищется такая элементарная посылка F, которая имеет неизвестное подвыражение T с тем же заголовком, что у выражения t. Проверяется, что все переменные списка R содержатся в T и что посылка F не имеет вида $T = a$, где a известно. Находятся такие термы Q, подстановка которых в t вместо переменных списка S дает выражение T. Определяются результаты $C_1, \ldots, C_m$ применения данной подстановки к антецедентам $A_1, \ldots, A_m$. Находится список D всех переменных кванторной приставки $x_1, \ldots, x_n$, входящих в утверждения $C_1, \ldots, C_m$, и проверяется, что он непуст. Решается вспомогательная задача на описание Z', условиями которой служит утверждения $C_1, \ldots, C_m$, посылками - все посылки текущей задачи на исследование, а неизвестными - переменные списка D. Эта задача имеет также цели "полный", "явное", "прямойответ", "и", "упростить". При обращении к ней устанавливается достаточно сильное ограничение на трудоемкость. Если в результате решения возникает система равенств, явно определяющих значения переменных списка D, то находится результат применения к консеквенту A_0 объединенной подстановки вместо переменных списков S, D. Этот результат регистрируется приемом как новое следствие, выведенное с помощью кванторной импликации. Он содержит ранее встречавшееся в задаче неизвестное выражение T и может оказаться полезным для дальнейшего. Уровень срабатывания приема равен 5.

Использование кванторной посылки задачи на описание для усмотрения истинности условия

Если решается задача на описание, в которой не требуется получить полный ответ, то кванторная импликация K в ее посылках может быть использована для усмотрения истинности элементарного условия F. Проверяется, что F является результатом подстановки p в консеквент импликации K некоторых термов вместо переменных кванторной приставки. После этого предпринимается цикл проверок истинности результатов применения той же подстановки p к антецедентам импликации. Для проверок используются проверочные операторы. В случае успеха условие F заменяется на константу "истина". Уровень срабатывания приема равен 5.

Использование равенства функциональной переменной константе

Если кванторная импликация имеет консеквент вида $f(t) = p$, где выражение t содержит переменные кванторной приставки, а выражение p - не содержит этих переменных, то предпринимается попытка заменить на p все представимые в виде $f(t)$ подвыражения, к контексту которых относится кванторная импликация. Перед каждой заменой проверяется истинность антецедентов в соответствующем подслучае. Уровень срабатывания приема равен 2.

Импликативное условие задачи на описание, не требующей полного ответа

Для устранения кванторных импликаций в условиях задачи на описание нет каких-то простых общих приемов. В задачах по элементарной алгебре они обычно исключаются за счет разрешения подкванторных утверждений относительно переменных кванторной приставки. Однако, в других разделах (например, в математическом анализе) такое разрешение редко бывает возможным. Чтобы избавиться от кванторной импликации, иногда удается использовать новые понятия, имеющие кванторные определения. Целесообразность перехода к таким понятиям целиком зависит от того, насколько развит аппарат решения задач, сформулированных на их языке.

Новые возможности для работы с кванторными импликациями в условиях задачи на описание появляются, если не требуется получить полный ответ. Тогда можно применять различные разновидности обратного вывода, подбирая такие бескванторные дополнительные условия задачи, при наличии которых консеквент импликативного условия становится следствием его антецедентов. Приведем несколько приемов, реализующих этот обратный вывод. Рассматриваемое импликативное условие обозначаем далее посредством $\forall_{x_1 \ldots x_n}(A_1 \,\&\ldots\&\, A_m \to A_0)$.

1. Попытка отождествления консеквента с антецедентом либо получения двух противоположных антецедентов. Пусть текущая задача на описание Z не имеет цели "полный" либо имеет цель "пример", причем консеквент A_0 и некоторый антецедент A_i элементарны. Если одно из утверждений A_0, A_i содержит неизвестные, а другое нет, причем то из них, которое "известно", представляет собой (с точностью до простейших преобразований) результат подстановки в "неизвестное" не содержащих переменных кванторной приставки термов $t_1, \ldots, t_k$ вместо неизвестных $y_1, \ldots, y_k$, то решается вспомогательная задача Z', полученная из Z заменой импликативного условия на $y_1 = t_1 \,\&\ldots\&\, y_k = t_k$. В случае успеха найденный ответ выдается как ответ на задачу Z.

   Аналогичным образом, рассматриваются два антецедента A_i, A_j, один из которых содержит неизвестные, а другой - нет. Если отрицание "известного" представляет собой результат подстановки в "неизвестное" не зависящих от переменных кванторной приставки термов $t_1, \ldots, t_k$ вместо неизвестных $y_1, \ldots, y_k$, то далее, как и выше, рассматривается вспомогательная задача Z'.

   Оба случая объединены в один прием, уровень срабатывания которого равен 9.

2. Исключение импликативного условия без неизвестных в антецедентах с помощью вспомогательной задачи на описание, имеющей цель "независит". Пусть все антецеденты $A_1, \ldots, A_m$ не содержат неизвестных текущей задачи на описание Z, а консеквент A_0 содержит. Если эта задача не имеет цели "полный" либо имеет цель "пример", то создается вспомогательная задача Z', посылками которой становятся все посылки задачи Z, пополненные утверждениями

$A_1, \ldots, A_m$. Условия ее получены заменой рассматриваемой кванторной импликации на группу конъюнктивных членов консеквента A_0. Цели задачи Z' воспроизводят цели задачи Z, с добавлением цели (независит $\ldots$), перечисляющей переменные $x_1, \ldots, x_n$ кванторной приставки. При необходимости перед созданием задачи Z' предпринимается переобозначение этих переменных. Требование независимости ответа R задачи Z' от переменных кванторной приставки, как легко видеть, гарантирует, что рассматриваемая кванторная импликация будет являться следствием R и посылок задачи Z. Поэтому R выдается в качестве ответа на задачу Z. Уровень срабатывания приема равен 6.

3. Попытка обратного вывода с помощью кванторной посылки. Как и выше, предполагаем, что текущая задача на описание не имеет цели "полный" либо имеет цель "пример". Пусть консеквент кванторной импликации имеет вхождения выражений вида "значение($f\ t$)", где f - переменная, не входящая в кванторную приставку. Рассматривается список S всех таких переменных f. Находится такая импликативная посылка $\forall_{y_1 \ldots y_p}(B_1 \ \& \ldots \& \ B_r \to B_0)$, что консеквент B_0 содержит все переменные $y_1, \ldots, y_p$ и имеет вхождение такой переменной списка S, которая не встречается в антецедентах $B_1, \ldots B_r$. Если существуют термы $t_1, \ldots, t_p$, подстановка которых в B_0 вместо переменных $y_1, \ldots, y_p$ дает консеквент A_0, то вводится вспомогательная задача Z', полученная из текущей заменой в рассматриваемом импликативном условии консеквента на конъюнкцию результатов подстановки $t_1, \ldots, t_p$ вместо $y_1 \ldots, y_p$ в антецеденты $B_1, \ldots, B_r$. Если на задачу Z' найден ответ, то он выдается как ответ на текущую задачу. Уровень срабатывания приема равен 6.

4. Использование вспомогательной задачи на описание, имеющей цель "длялюбого". Ранее уже рассматривался прием, вводящий для исключения импликативного условия вспомогательную задачу Z' на описание с целью "независит". Посылками ее служили все утверждения из контекста импликативного условия (т.е. посылки и остальные условия), а также антецеденты; условиями являлись конъюнктивные члены консеквента. При получении ответа R, не зависящего от переменных кванторной приставки, можно было предпринять попытку замены импликативного условия на этот ответ. Часто бывает необходима более общая схема рассуждений, при которой по ходу решения задачи Z' разрешается вводить вспомогательные объекты, существование которых, вообще говоря, не вытекает из ее исходных посылок, но накладывает не слишком сильные дополнительные ограничения на параметры текущей задачи Z. Эти ограничения (разумеется, не зависящие от переменных кванторной приставки) присоединяются по окончании решения к ответу R. Чтобы указать на данный режим решения, задача Z' снабжается целью "длялюбого".

   Прием, вводящий задачу Z', сначала находит список S всех свободных переменных f импликативного условия, имеющих вхождение выражения $f(t)$, некоторые переменные которого связаны внешними кванторами и описателями. Проверяется, что все переменные списка S появляются только в консеквенте импликативного условия. Это - эвристический фильтр, отсекающий ситуации, не типичные для задач, в которых прием был использован. Далее, как указано выше, создается задача Z'. Она имеет лишь три цели - "прямойответ", "длялюбого" и "независит $x_1 \ldots x_n$", указывающей на переменные кванторной приставки.

   В процессе решения данной задачи создаются комментарии (блокпосылок Q_1

Q_2), регистрирующие ввод вспомогательных объектов. Здесь Q_1 - набор троек $(X_1\ X_2\ P)$, соответствующих дополнительным посылкам "существует$(X\ B)$", вводящим вспомогательные объекты X. Связывающая приставка X разбивается на две части - X_1 и X_2. Переменные первой части считаются не зависящими от переменных кванторной приставки $x_1, \ldots, x_n$ рассматриваемого импликативного условия; переменные второй части - зависящими от них. P - набор конъюнктивных членов утверждения B. Квантор существования в дополнительной посылке устраняется специальным приемом, после чего все переменные списка X становятся обозначениями некоторых объектов, рассматриваемых в задаче Z'. Решение о том, какие переменные следует относить к X_1, а какие - к X_2, принимается на основе эвристических соображений. Логическая корректность последующих преобразований основана на учете этого решения. Q_2 есть набор конъюнктивных членов результата упрощения утверждения, выражающего, что для любых удовлетворяющих антецедентам $A_1, \ldots, A_m$ значений переменных $x_1, \ldots, x_n$ существуют такие значения переменных из списков X_2, представленных в Q_1, что истинна конъюнкция всех утверждений P. Обычно вводятся такие вспомогательные объекты, чтобы утверждения в Q_2 не имели вида кванторных импликаций. Задача Z' может иметь несколько различных комментариев (блокпосылок ...), относящихся к несвязанным между собой группам вспомогательных объектов.

По окончании решения задачи Z' составляется объединенный список D конъюнктивных членов ее ответа и всех утверждений, содержащихся в списках Q_2 комментариев "блокпосылок". На конъюнкцию элементов списка D навешивается квантор существования по всем переменным списков X_1. Пусть H - результирующее утверждение. Как нетрудно видеть, рассматриваемое импликативное условие является следствием этого утверждения относительно своего контекста, и далее можно предпринять попытку решения вспомогательной задачи Z'', полученной из текущей задачи заменой импликативного условия на H.

В приеме предусмотрено особое рассмотрение случая, когда все неизвестные текущей задачи являются несущественными. Например, так бывает при сведении к задаче на описание задачи на доказательство существования. Здесь предпринимается попытка решения указанной выше задачи Z'' с ослабленными целями, разрешающими выдачу ответа M просто по исчерпании средств (как конъюнкцию текущих условий). В случае, когда M оказалось константой "истина", выдается ответ "истина". Иначе - рассматривается отрицание результата навешивания на утверждение M квантора существования по всем входящим в него неизвестным, которое регистрируется в списке посылок текущей задачи. Смысл такого преобразования состоит в отсечении ситуации, для которой задачу удалось решить, так что дальнейшие действия относятся к альтернативным ситуациям.

Уровень срабатывания приема равен 7.

В качестве примера применения данного приема рассмотрим задачу на доказательство равенства предела суммы двух сходящихся числовых последовательностей сумме их пределов. После несложных преобразований приходим к задаче на описание, имеющей следующие посылки:

последовательность(f, R), последовательность(g, R), $\forall_\epsilon(0 < \epsilon\ \&\ \epsilon -$ число $\rightarrow$ $n(\epsilon) -$ натуральное), $\forall_{\epsilon k}(0 < \epsilon\ \&\ \epsilon -$ число $\&\ k -$ натуральное $\&\ n(\epsilon) \leq k \rightarrow$

$|a - f(k)| < \epsilon)$, $\forall_\epsilon(0 < \epsilon \ \& \ \epsilon - \text{число} \to m(\epsilon) - \text{натуральное})$, $\forall_{\epsilon k}(0 < \epsilon \ \& \ \epsilon - \text{число} \ \& \ k - \text{натуральное} \ \& \ m(\epsilon) \leq k \to |b - g(k)| < \epsilon)$, $0 < c$, $c -$ число.

Условиями задачи служат кванторная импликация $\forall_N(N - \text{натуральное} \ \& \ p \leq N \to |a + b - f(N) - g(N)| < c$ и утверждение "p - натуральное". Задача имеет единственную несущественную неизвестную p.

Переходя к вспомогательной задаче Z', добавляем к указанным выше посылкам утверждения "p-натуральное", "N-натуральное" и $p \leq N$. Условием этой задачи служит единственное утверждение $|a + b - f(N) - g(N)| < c$.

Так как ответ задачи не должен зависеть от переменной N, необходимо как-то исключить из условия выражения $f(N)$ и $g(N)$. Сделать это можно только с помощью информации, содержащейся в импликативных посылках для f и g. Однако, чтобы воспользоваться данными посылками, нужно обеспечить истинность их антецедентов. Например, в случае f нужно рассмотреть такое число ϵ_1, что истинны утверждения $0 < \epsilon_1$, $\epsilon_1 -$ число, $n(\epsilon_1) \leq N$. Вообще говоря, ниоткуда не следует существование этого числа. Однако, рассмотрение его необходимо, и соответствующий прием (мы рассмотрим его позднее) введет указанные утверждения в качестве дополнительных посылок, сопроводив их комментарием (блокпосылок Q_1 Q_2). Здесь Q_1 состоит из единственной тройки (ϵ_1, $\emptyset$, $\{0 < \epsilon_1, \epsilon_1 - \text{число}, n(\epsilon_1) \leq N\}$). Для получения Q_2 предпринимается упрощение утверждения $\forall_N(N - \text{натуральное} \ \& \ p \leq N \to (0 < \epsilon_1 \ \& \ \epsilon_1 - \text{число} \ \& \ n(\epsilon_1) \leq N))$. В результате Q_2 оказывается состоящим из $0 < \epsilon_1$, $\epsilon_1 -$ число, $n(\epsilon_1) \leq p$. Аналогичным образом, для g получаем комментарий (блокпосылок ...), у которого Q_2 состоит из $0 < \epsilon_2$, $\epsilon_2 -$ число, $m(\epsilon_2) \leq p$. Используя дополнительные посылки и кванторные импликации в основных посылках, далее выводим следствия $|a - f(N)| < \epsilon_1$, $|b - g(N)| < \epsilon_2$. Это позволяет получить оценку $\epsilon_1 + \epsilon_2$ выражения $|a + b - f(N) - g(N)|$ из условия задачи. Так как данная оценка не зависит от N, неравенство $\epsilon_1 + \epsilon_2 < c$ выдается в качестве ответа на Z'. Далее добавляем к нему все утверждения из накопителей Q_2 и навешиваем квантор существования по ϵ_1, ϵ_2. Получаем утверждение $\exists_{\epsilon_1\epsilon_2}(0 < \epsilon_1 \ \& \ \epsilon_1 - \text{число} \ \& \ n(\epsilon_1) \leq p \ \& \ 0 < \epsilon_2 \ \& \ \epsilon_2 - \text{число} \ \& \ m(\epsilon_2) \leq p \ \& \ 0 < c - \epsilon_1 - \epsilon_2)$. Оно и заменяет рассматриваемое импликативное условие. Квантор общности оказался устраненным, и далее задача легко доводится до ответа.

Накопление необходимых условий для получения достаточного условия

Если кванторная импликация является условием задачи на описание, в которой требуется получить полный ответ, то имеется еще одна возможность исключения квантора общности. Она состоит в накоплении такого списка бескванторных следствий данного условия, который в итоге оказывается эквивалентным самому условию. Прием, реализующий эти действия, прежде всего проверяет целесообразность попытки исключения квантора. Например, отсекаются ситуации, когда импликация входит в ответ как серия ограничений на допустимые значения неизвестных. После этого выполняется быстрая проверка реализуемости антецедентов в контексте рассматриваемого условия. Для вывода следствий из кванторной импликации используется вспомогательная задача на исследование Z'. Ее посылками служат все утверждения из контекста импликации, пополненные антецедентами и конъюнктивными членами консеквента. Задача имеет цель "длялюбого", а также цель, перечисляющую неизвестные текущей задачи на описание Z. Для использования в задаче Z отбираются лишь такие посылки задачи Z', которые не зависят от переменных кванторной приставки импликативного условия. Они регистрируются в комментарии (полныепосылки ...). В нем же регистрируются бескванторные результаты упрощения кванторных импликаций, полученных из рассматриваемого импликативного условия заменой консеквента на выведенные посылки задачи Z', зависящие от кванторной приставки. Это обеспечивается специальным приемом символа "исследовать" (см. выше). По завершении решения задачи Z' рассматривается содержимое R накопителя (полныепосылки ...). Если удается доказать, что импликативное условие является следствием своего контекста Q, пополненного утверждениями списка R, то эти утверждения упрощаются относительно Q, и предпринимается замена рассматриваемого условия задачи Z на их конъюнкцию. Уровень срабатывания приема равен 5.

В качестве примера применения данного приема рассмотрим задачу нахождения множества частичных пределов последовательности $1/2, 1/3, 2/3, 1/4, 2/4, 3/4, \dots$. После простейших преобразований она сводится к задаче на описание, имеющей условие $\forall_{an}(0 < a \ \& \ a -$ число $\& \ n -$ натуральное $\rightarrow \exists_{bm}(n < m \ \& \ |x - \frac{b}{m+1}| < a \ \& \ b \in \{1, \dots, m\} \ \& \ m -$ натуральное$))$. Единственной неизвестной является переменная x. Вспомогательная задача на исследование, создаваемая приемом, имеет посылки $x -$ число, $0 < a$, $a -$ число, $n -$ натуральное, $\exists_{bm}(n < m \ \& \ |x - \frac{b}{m+1}| < a \ \& \ b \in \{1, \dots, m\} \ \& \ m -$ натуральное$)$. Квантор существования в последней посылке устраняется специальным приемом, заменяющим переменные b, m на новые переменные c, d. При этом возникают посылки $n < d$, $|x - \frac{c}{d+1}| < a$, $c \in \{1, \dots, d\}$, $d -$ натуральное и добавляется цель (независит c d). Из неравенства с модулем выводятся следствия $-a + \frac{c}{d+1} < x$, $x < a + \frac{c}{d+1}$. Чтобы получить неравенства для x, не содержащие переменных c, d, применяются приемы, обращающиеся к синтезаторам "верхняяоценка" и "нижняяоценка". Они выводят следствия $-a < x$, $x < a + 1$. Сразу по получении каждого из них применяется общий прием символа "исследовать". В первом случае он упрощает утверждение $\forall_a(0 < a \ \& \ a -$ число $\rightarrow -a < x)$ и получает $0 \leq x$, во втором - упрощает утверждение $\forall_a(0 < a \ \& \ a -$ число $\rightarrow x < a + 1$ и получает $x \leq 1$. Оба результата регистрируются в комментарии (полныепосылки ...). По окончании решения вспомогательной задачи на исследование предпринимается попытка доказать, что из неравенств $0 \leq x, x \leq 1$ вытекает истинность рассматриваемого импликативного условия. Эта попытка оказывается успешной, и прием заменяет импликативное условие на конъюнкцию неравенств.

Подбор контрпримера к импликативной посылке задачи на исследование, имеющей цель "противоречие"

Задача на исследование, имеющая цель "противоречие", обычно решается для доказательства какого-либо утверждения от противного. При ее решении нужно усмотреть противоречивость посылок, т.е. вывести следствие "ложь". Если кванторная импликация $\forall_{x_1 \dots x_n}(A_1 \ \& \ \dots \ \& \ A_m \rightarrow A_0)$ встречается в посылках такой задачи, то можно попытаться доказать ее отрицание, подобрав контрпример. С этой целью создается вспомогательная задача на описание Z', условиями которой становятся все антецеденты $A_1, \dots, A_m$ и отрицание консеквента A_0. Посылками служат все остальные посылки текущей задачи на исследование. Цели вспомогательной задачи суть (неизвестные $x_1 \dots x_n$), (параметры $x_1 \dots x_n$), "полный", "пример", "отрицание". Последняя цель просто служит пометкой, указывающей на источник появления данной задачи. Максимальный уровень обращения к задаче Z' невелик и устанавливается из эвристических соображений. Если задача Z' решена, то импликативная посылка текущей задачи на исследование заменяется на константу "ложь". Уровень срабатывания приема равен 5.

Попытка вывода следствий из консеквента кванторной импликации для последующей кванторной свертки

Если некоторая задача имеет посылку P вида $\forall_{x_1\ldots x_n}(A_1\ \&\ \ldots\ \&\ A_m \rightarrow A_0)$, то для получения бескванторных следствий может быть использована следующая процедура. Создается вспомогательная задача на исследование Z', посылками которой становятся все отличные от P посылки текущей задачи, антецеденты $A_1,\ldots,A_m$ и консеквент A_0. В задаче Z' предпринимается вывод следствий, причем лимит времени, отпущенного на ее решение, не очень велик. По окончании вывода следствий рассматриваются те посылки Q задачи Z', отличные от ее исходных посылок, которые получены с существенным использованием консеквента A_0. Последнее определяется по комментарию (выводимо ...) к посылке Q. Далее решается задача на упрощение кванторной импликации $\forall_{x_1\ldots x_n}(A_1\ \&\ \ldots\ \&\ A_m \rightarrow Q)$, результатом которой служит некоторое утверждение R. Если R бескванторное, то оно регистрируется в посылках текущей задачи как следствие посылки P. В комментарии (выводимо ...) к R учитываются также другие посылки, использованные при выводе. Уровень срабатывания приема равен 5.

Вывод следствий из кванторной посылки типа "для любого существует" с помощью вспомогательной задачи на описание

Приведем еще один прием вывода следствий из кванторных импликаций, на этот раз имеющих уже специальный вид - $\forall_{x_1\ldots x_n}(A_1\ \&\ \ldots\ \&\ A_m \rightarrow \exists_{y_1\ldots y_k}(B_1\ \&\ \ldots\ \&\ B_p))$. Прием применяет схему рассуждений "от противного". Сначала рассматривается отрицание N кванторной импликации. Оно имеет вид

$$\exists_{x_1\ldots x_n}(A_1\ \&\ \ldots\ \&\ A_m\ \&\ \forall_{y_1\ldots y_k}(B_1\ \&\ \ldots\ \&\ B_{p-1} \rightarrow \neg B_p)).$$

Чтобы получить бескванторное утверждение P, следствием которого являлась бы кванторная импликация $\forall_{y_1\ldots y_k}(B_1\ \&\ \ldots\ \&\ B_{p-1} \rightarrow \neg B_p)$, решается вспомогательная задача на описание, имеющая цель "длялюбого". Как и в рассмотренном ранее приеме, она использует комментарий (блокпосылок ...), накапливающий дополнения к ответу. Далее рассматриваются утверждения $A_1,\ldots,A_m,P$, которые становятся условиями еще одной вспомогательной задачи на описание. В ней требуется подобрать пример для неизвестных $x_1,\ldots,x_n$, причем эти неизвестные объявляются несущественными. Как легко видеть, N является следствием ответа R на указанную задачу. Эта задача имеет цель "попыткаспуска", т.е. по исчерпании средств в качестве ответа выдается конъюнкция ее текущих условий. На оставшиеся неисключенными неизвестные в R навешивается квантор существования. Завершает рассуждения "от противного" присоединение отрицания утверждения R в качестве новой посылки. Уровень срабатывания приема равен 8.

Данный прием возник при рассмотрении задачи на доказательство существования наибольшего либо наименьшего элемента у сходящейся числовой последовательности. Эту задачу можно найти в разделе задачника "Математический анализ" - "Задачи на доказательство" - "Последовательности" и проследить по ней ход применения приема. В некоторый момент решения возникает задача на описание, имеющая посылки $\forall_n(n$ – натуральное $\rightarrow f(n) \leq d)$, $\forall_c(0 < c\ \&\ c$ – число $\rightarrow$

$p(c)$ – натуральное), $\forall_m(m$ – натуральное $\to \exists_k(f(k) < f(m)$ & k – натуральное)), d – число, последовательность(f, R), $\forall_{cl}(0 < c$ & c – число & l – натуральное & $p(c) \le l \to |d - f(l)| < c)$. Прием применяется к посылке $\forall_m(m$ – натуральное $\to \exists_k(f(k) < f(m)$ & k – натуральное)). Отрицание ее консеквента имеет вид $\forall_k(k$ – натуральное $\to f(m) \le f(k))$. Создается задача на описание с целью "длялюбого", посылки которой суть: k – натуральное, m – натуральное, $\forall_n(n$ – натуральное $\to f(n) \le d)$, $\forall_c(0 < c$ & c – число $\to p(c)$ – натуральное), последовательность(f, R), d – число, $\forall_c l(0 < c$ & c – число & l – натуральное & $p(c) \le l \to |d - f(l)| < c)$. Условием ее служит неравенство $f(m) \le f(k)$. На эту задачу решатель получает ответ $f(m) \le d - a$ & $f(m) \le \inf(\mathrm{set}_x(\exists_i(i \in \{1, \dots, p(a) - 1\}$ & $x = f(i))))$. Здесь a - вспомогательный параметр, условия на который извлекаются из комментария (блокпосылок ...). После учета условий на a ответ преобразуется к виду $\exists_a(a$ – число & $0 < a$ & $f(m) \le d - a$ & $f(m) \le \inf(\mathrm{set}_x(\exists_i(i \in \{1, \dots, p(a) - 1\}$ & $x = f(i)))))$. Далее решается задача на подбор значения m, удовлетворяющего предыдущему утверждению и условию "m – натуральное". По исчерпании средств, на нее выдается ответ $\exists_i(f(i) < d$ & i – натуральное). Отрицание последнего утверждения, имеющее вид $\forall_i(i$ – натуральное $\to d \le f(i))$, заносится в качестве новой посылки задачи. Так как в посылках имеется встречное кванторное неравенство для f, обнаруживается, что последовательность константная, и далее задача быстро доводится до конца.

Использование индуктивного предположения, имеющего вид кванторной импликации

Приемы доказательства по индукции создают вспомогательные задачи, списки посылок которых пополняются утверждениями - конъюнктивными членами индуктивного предположения. Вывод следствий из этих утверждений является приоритетным, и для усиления его создан ряд дополнительных приемов. Один из них относится к случаю, когда индуктивное предположение имеет вид кванторной импликации $\forall_{x_1 \dots x_n}(A_1$ & ... & $A_m \to A_0)$. Проверяется, что переменная либо выражение, по которому ведется индукция, встречается в антецедентах. Находится элементарный антецедент A_i, содержащий все переменные кванторной приставки, и для него выбирается посылка P, представляющая собой результат применения к A_i некоторой подстановки s вместо переменных кванторной приставки. Проверяется, что применение s ко всем оставшимся антецедентам дает утверждения, истинность которых относительно контекста быстро устанавливается с помощью процедуры "извлекается". Тогда находится результат применения s к консеквенту A_0, и этот результат регистрируется в посылках текущей задачи. Уровень срабатывания приема равен 3.

Использование кванторной импликации с функциональными переменными для вывода следствий в задаче на исследование, имеющей цель "противоречие"

Если кванторная импликация $\forall_{x_1 \dots x_n}(A_1$ & ... & $A_m \to A_0)$ имеет для каждой переменной x_i кванторной приставки вхождение в консеквент выражения вида $f(x_i)$, то можно попробовать использовать ее для вывода утверждения, содержащего уже встречавшиеся в задаче выражения вида $g(t)$. Такое утверждение, дающее дополнительную связь между уже имеющимися объектами, может оказаться полезным для усмотрения противоречия. Прием определяет для каждой переменной x_i список T_i таких выражений t, что консеквент A_0 содержит $f(x_i)$, а некоторая элементарная посылка - $f(t)$. Далее перебираются всевозможные наборы $(t_1, \ldots t_n)$ выражений t_i из списков T_i. Предварительно проверяется. что число этих наборов невелико (для рассматривавшихся примеров оказалось достаточно верхней границы числа наборов, равной 3). Как только обнаруживается набор, подстановка которого вместо переменных кванторной приставки в антецеденты дает утверждения, усматриваемые с помощью проверочных операторов, выполняется вывод результата применения той же подстановки к консеквенту A_0. Уровень срабатывания приема равен 3.

1.9 Приемы символа "существует"

Хотя квантор существования с помощью отрицания выражается через квантор общности, лишь простейшие его приемы аналогичны рассмотренным выше. Заметим также, что корневые вхождения квантора существования обычно устраняются с помощью несложных логических переходов - в посылках за счет ввода вспомогательных объектов, в условиях - за счет перехода к задаче на описание либо за счет ввода дополнительных неизвестных такой задачи.

Простейшая стандартизация

Перечислим приемы общей стндартизации кванторов существования. В основном, они аналогичны таким приемам, введенным для квантора общности. Уровень срабатывания их равен 0.

1. Переобозначение связанных переменных. Переменные кванторной приставки переобозначаются так, чтобы они отличались от свободных переменных утверждений того контекста, относительно которого рассматривается квантор существования. Для этого используется процедура "нормализациясвязок".

2. Исключение из кванторной приставки переменных, не имеющих свободных вхождений в подкванторное утверждение.

3. Устранение из кванторной приставки повторных вхождений одной и той же переменной.

4. Слияние кванторных приставок вложенных кванторов существования. Выполняется замена утверждения $\exists_{x_1 \ldots x_n}(\exists_{y_1 \ldots y_m} A)$ на $\exists_{x_1 \ldots x_n y_1 \ldots y_m} A$.

5. Устранение дизъюнкции под квантором существования. Выполняется замена утверждения $\exists_{x_1 \ldots x_n}(A_1 \vee \ldots \vee A_m)$ на $\exists_{x_1 \ldots x_n} A_1 \vee \ldots \vee \exists_{x_1 \ldots x_n} A_m$.

6. Подстановка явно определенного равенством значения связанной переменной. Утверждение $\exists_{x_1 \ldots x_n}(x_i = t \;\&\; A)$, где терм t не зависит от x_i, заменяется на $\exists_{x_1 \ldots x_{i-1} x_{i+1} \ldots x_n} B$. Здесь B - результат подстановки выражения t вместо x_i в утверждение A. На применение этого преобразования имеется ряд ограничений, связанных с теми случаями, когда равенство под квантором существования нужно для задания некоторого параметрического описания.

Разбиение множества конъюнктивных членов подкванторного утверждения на компоненты связности по зависимости от переменных кванторной приставки

Конъюнктивные члены $A_1, \ldots, A_n$ подкванторного утверждения в кванторе существования $\exists_{x_1 \ldots x_m}(A_1 \ \& \ \ldots \ \& \ A_n)$ разбиваются на наибольшее число таких непересекающихся подгрупп $B_1, \ldots, B_k$, что утверждения из двух различных подгрупп не зависят от одной и той же переменной кванторной приставки. Если $k > 1$, то рассматриваются конъюнкции $C_1, \ldots, C_k$ утверждений групп $B_1, \ldots, B_k$, и квантор существования заменяется на конъюнкцию кванторов $\exists_{X_1} C_1$, ... , $\exists_{X_k} C_k$. Здесь X_i - все переменные $x_1, \ldots, x_n$, имеющие свободные вхождения в утверждение C_i. Если какая - либо группа переменных X_i пуста, то C_i берется без навешивания на него квантора существования. В зависимости от контекста, прием может срабатывать на уровнях 0, 1 и 3.

Устранение дизъюнкции, являющейся конъюнктивным членом утверждения под квантором существования

Утверждение $\exists_{x_1 \ldots x_n}(A \ \& \ (B1 \vee \ldots \vee B_m))$ заменяется на $\exists_{x_1 \ldots x_n}(A \ \& \ B_1) \vee \ldots \vee \exists_{x_1 \ldots x_n}(A \ \& \ B_m)$. В зависимости от контекста, уровень срабатывания приема равен 1 либо (в редких случаях)4. При редактировании ответа задачи на описание прием обычно блокируется.

Группировка кванторов наружу

Если конъюнктивный член подкванторного утверждения сам является квантором существования, то этот квантор выносится наружу: утверждение $\exists_{x_1 \ldots x_n}(A \ \& \ \exists_{y_1 \ldots y_m} B)$ заменяется на $\exists_{x_1 \ldots x_n y_1 \ldots y_m}(A \ \& \ B)$. Уровень срабатывания приема равен 1. Заметим, что к моменту применения данного приема уже выполнено переобозначение связанных переменных $y_1, \ldots, y_m$, гарантирующее, что они не имеют свободных вхождений в утверждение A.

Решение задачи на доказательство существования путем подбора примера

Если условием задачи на доказательство служит утверждение $\exists_{x_1 \ldots x_n}(A_1 \ \& \ldots \ \& \ A_m)$, $m \geq 1$, то создается вспомогательная задача на описание Z', имеющая тот же список посылок и условия $A_1, \ldots, A_m$. Эта задача имеет цели "полный", "пример", (неизвестные $x_1 \ldots x_n$), (параметры $x_1 \ldots x_n$). Комментарии к ее посылкам переносятся из текущей задачи с помощью справочника "существует". Если на задачу Z' получен ответ, отличный от символов "отказ" и "ложь", то на текущую задачу выдается ответ "истина". В противном случае решение текущей задачи на доказательство продолжается с привлечением других средств. Уровень срабатывания приема равен 2.

Элиминация квантора существования в посылках за счет ввода вспомогательных объектов

Если в посылках задачи на доказательство либо задачи на описание, не имеющей цели "прямойответ", встречается утверждение вида $\exists_{x_1 \ldots x_n} A$, то квантор существования устраняется. Здесь рассматриваются два случая:

1. Текущая задача - на доказательство либо не имеет цели (независит $z_1 \dots z_m$), переменные которой имеют свободные вхождения в рассматриваемую кванторную посылку. Тогда выбираются не использованные в задаче переменные $y_1, \dots, y_n$, находится результат B переобозначения в A переменных x_i на y_i, и кванторная посылка заменяется на B.

2. Текущая задача - на описание и имеет цель (независит $z_1 \dots z_m$), переменные которой имеют свободные вхождения в рассматриваемую посылку. Находится список $Z_1, \dots, Z_p$ всех таких переменных. Выбираются не использованные в задаче переменные $y_1, \dots, y_n$ и определяется результат B подстановки в A вместо переменных $x_1, \dots, x_n$ выражений $y_1(Z_1 \dots\ Z_p), \dots, y_n(Z_1 \dots\ Z_p)$. Затем кванторная посылка заменяется на B.

В обоих случаях корректируется комментарий (новаяпеременная $S_1\ S_2$) к посылкам текущей задачи. Такой комментарий перечисляет в наборе S_1 все вспомогательные переменные посылок, возникшие в процессе вывода следствий. Список S_2 имеет ту же длину, что и список S_1. На позиции его, соответствующей переменной x списка S_1, указывается набор ранее введенных вспомогательных переменных, от которых x может зависеть. В данном приеме происходит добавление к списку S_1 переменных $y_1, \dots, y_n$ и указание для них в S_2 пересечения S_1 со списком параметров кванторной посылки. Уровень срабатывания приема равен 2.

Элиминация квантора существования в посылках задачи на исследование, имеющей цель "длялюбого"

Этот прием аналогичен предыдущему, но относится к задачам на исследование. Требуется также, чтобы задача имела цель "длялюбого" (см. приведенный ранее прием, анализирующий кванторную импликацию в условиях задачи на описание). Для рассматриваемой посылки $\exists_{x_1 \dots x_n} A$ выбираются новые переменные $y_1, \dots, y_n$. Если задача имела цель (независит $\dots$), то к списку переменных этой цели присоединяются переменные $y_1, \dots, y_n$. В противном случае создается цель (независит $y_1 \dots y_n$). Далее посылка заменяется на результат переобозначения в A переменных x_i на y_i. Уровень срабатывания приема равен 2.

Ввод сколемовских функций для элиминации квантора существования в консеквенте кванторной посылки

Квантор существования исключается не только в ситуациях, когда он является заголовком посылки, но и в ситуациях, когда он является заголовком консеквента посылки - кванторной импликации. Именно, если посылка задачи на доказательство либо задачи на описание, не имеющей цели "прямойответ", представима в виде $\forall_{x_1 \dots x_n}(A_1\ \& \dots\ \&\ A_m \to \exists_{y_1 \dots y_m} A_0)$, то выбираются новые переменные $f_1, \dots, f_m$, и посылка заменяется на $\forall_{x_1 \dots x_n}(A_1\ \& \dots\ \&\ A_m \to B$. Здесь B - результат подстановки в утверждение A_0 выражений $f_1(x_1 \dots x_n), \dots, f_n(x_1 \dots x_n)$ вместо переменных $y_1, \dots, y_m$. Прием корректирует комментарий (новаяпеременная $S_1\ S_2$) так же, как это делал предыдущий прием. Переменные $f_1, \dots, f_m$, возникающие при данном преобразовании, в математической логике обычно называются "сколемовскими функциями". Прием пока реализован лишь для случая отсутствия у задачи на описание цели (независит $\dots$), пересекающейся с параметрами посылки. Уровень его срабатывания равен 4.

Элиминация условия существования в задаче на описание, решаемой с целью "пример"

Если задача на описание имеет условие $\exists_{x_1\ldots x_n}A$, то в ряде случаев можно заменить данное условие на A, присоединив переменные $x_1,\ldots,x_n$ к списку неизвестных задачи и указав на то, что они являются несущественными. Это делается, если задача имеет цель "пример", либо цель "замещение" (переформулировка условий без использования неизвестных, которые в таких задачах могут встречаться и в посылках), либо цель "функционально" (в задаче нужно лишь установить факт однозначного определения значений неизвестных по значениям известных параметров). Кроме того, срабатывание приема допускается при отсутствии неизвестных либо при наличии у условия комментария "параметры". Прием блокируется на этапе редактирования ответа. Если задача имела цель (независит $y_1\ldots y_k$), то для каждой переменной x_i вводится дополнительная цель (сокращнеизв x_i y_1 $\ldots$ y_k), разрешающая неизвестной x_i зависеть от переменных $y_1,\ldots,y_k$. Прием не преобразует указанным выше образом текущую задачу, а вводит и решает вспомогательную задачу. Если на нее получен "отказ", то и на текущую задачу выдается "отказ". В противном случае - в зависимости от целевой установки либо сразу выдается найденный ответ, либо на него навешивается квантор существования по переменным $x_1,\ldots,x_n$, и ответ выдается лишь после упрощения этого квантора. Для упрощения используется еще одна вспомогательная задача на описание, снабженная целью "редакция". Уровень срабатывания приема равен 3.

Удаление посылки "существует(...)" за счет непосредственного подбора значений

Если посылка задачи имеет вид $\exists_{x_1\ldots x_n}(A_1 \,\&\, \ldots \,\&\, A_m)$, где $A_1,\ldots,A_m$ - элементарные утверждения, то предпринимается попытка усмотреть среди прочих посылок такие элементарные утверждения $B_1,\ldots,B_m$, которые являются результатами подстановки вместо переменных $x_1,\ldots,x_n$ в утверждения $A_1,\ldots,A_m$ некоторых выражений $t_1,\ldots,t_n$. Для этого используется процедура "подборпосылок". Если она выдает искомый результат, то рассматриваемая кванторная посылка удаляется. Уровень срабатывания приема равен 1.

Разрешение подкванторного условия относительно переменных кванторной приставки

Аналогичный прием уже рассматривался для кванторной импликации. Проверяется отсутствие целей и комментариев, блокирующих попытку разрешения подкванторного утверждения условия $\exists_{x_1\ldots x_n}A$ относительно неизвестных $x_1,\ldots,x_n$. Создается вспомогательная задача Z', посылками которой служат все утверждения из контекста рассматриваемого условия, а условиями - конъюнктивные члены утверждения A. Цели задачи суть "полный", "явное", "прямойответ", (неизвестные $x_1\ldots x_n$), (параметры $x_1\ldots x_n$), а также ряд целей, переносимых из текущей задачи. Если при решении задачи Z' получен ответ B, отличный от A, то происходит замена условия на $\exists_{x_1\ldots x_n}B$. Смысл преобразования состоит в том, что если подкванторное утверждение разрешено относительно переменных кванторной приставки, то обычно удается вообще исключить квантор, используя простейшие эквивалентные преобразования. Уровень срабатывания приема равен 3.

Перенесение неизвестной посылки "существует(...)" из блока анализа во внешнюю задачу на описание

Аналогичный прием переносил из блока анализа во внешнюю задачу дизъюнкцию, содержащую неизвестные. Собственно перенесение квантора существования выполняется процедурой "замещениеусловий", которая пытается дополнить это условие необходимыми сопровождающими утверждениями, извлекаемыми из посылок задачи на исследование, а при возможности - также отбросить условия внешней задачи, оказывающиеся после перенесения избыточными. Уровень срабатывания приема равен 2. Так как появление в посылках блока анализа дизъюнкции либо утверждения существования, содержащих неизвестные, приводит к почти немедленному перенесению их во внешнюю задачу, сопровождаемому для дизъюнкции разбором случаев, а для квантора - попытками его исключения, то нужно вводить новые посылки одного из указанных видов лишь при наличии достаточно веских причин.

Устранение сдвоенных переменных кванторной приставки

Реализована несколько ослабленная версия приема, имевшегося для кванторных импликаций. Рассматривается утверждение вида $\exists_{x_1 \ldots x_n}(A_1 \ \& \ \ldots \ \& \ A_m)$. Если для двух различных переменных кванторной приставки x_i, x_j некоторые A_k, A_p имеют вид $P(x_i)$, $P(x_j)$, где P - название типа объектов, причем во всех остальных конъюнктивных членах подкванторного утверждения эти переменные встречаются только под знаком одной и той же операции f (в некоммутативном случае - в одном и том же порядке), то предпринимается склейка данных переменных. Предварительно проверяется, что f имеет единицу - это гарантирует, что на значения новой переменной не возникнут дополнительные ограничения. Фактически в качестве новой переменной берется сама переменная x_i: выражения $f(x_i, x_j)$ заменяются на x_i, а $P(x_j)$ отбрасывается.

Параметрические описания, определяющие значения неизвестных задачи

В ответах задач на описание могут появляться утверждения вида $\exists_{x_1 \ldots x_n}(P \ \& \ y_1 = t_1 \ \& \ \ldots y_n = t_n)$, где $y_1, \ldots, y_n$ - неизвестные; P - утверждение без неизвестных, ограничивающее множество допустимых значений переменных $x_1, \ldots, x_n$; $t_1, \ldots, t_n$ - выражения без неизвестных. Такие утверждения называются явными параметрическими описаниями множества допустимых значений неизвестных $y_1, \ldots, y_n$; переменные $x_1, \ldots, x_n$ суть параметры описания. Иногда рассматриваются также неявные параметрические описания, у которых значения неизвестных $y_1, \ldots, y_n$ не задаются явными выражениями через параметры, а связываются с ними каким-либо косвенным образом. Например, могут указываться промежутки для значений неизвестных. Чтобы усматривать и упрощать условия задачи, представляющие собой параметрические описания, а также чтобы непосредственно усматривать готовый ответ, составленный из таких описаний, служат специальные приемы. Перечислим те из них, которые реализованы на ЛОСе:

1. Усмотрение параметрического описания серии значений неизвестных.

   Если условие задачи представляет собой параметрическое описание значений неизвестных либо введено для последующего преобразования к такому описанию, то оно помечается комментарием "серия". Часто комментарии "серия" вводятся приемами, создающими кванторное условие. При отсутствии данного комментария у условия вида $\exists_{x_1 \ldots x_n} A$ предпринимается попытка усмотреть в нем параметрическое описание. Прежде всего, проверяется, что задача на описание не имеет целей "редакция" и "пример". Затем проверяется, что утверждение A имеет вид $y_1 = t_1 \ \& \ \ldots \ \& \ y_k = t_k \ \& \ B$, где $y_1, \ldots, y_k$ - неизвестные; $t_1, \ldots, t_k, B$ не содержат неизвестных. Проверяется, что каждое вхождение в задачу неизвестной y_i, $i \in \{1, \ldots, k\}$, не относящееся к рассматриваемому кванторному условию, относится к простейшему указателю типа данных $P(y_i)$. Проверяется отсутствие у B конъюнктивных членов с заголовками "равно", "число". После этого кванторное условие сопровождается комментарием "серия". Уровень срабатывания приема - 2 либо 3.

2. Переобозначение связанных переменных параметрического описания.

   Переменные кванторной приставки условия $\exists_{x_1 \ldots x_n} A$ переобозначаются так, чтобы они не имели свободных вхождений в утверждения контекста данного условия. Это делает на уровне 0 прием, о котором уже говорилось ранее. Иногда бывает необходимо дополнительное переобозначение. Если задача на описание Z имеет комментарий (контекст Z' B), то при редактировании ее ответа все утверждения списка B будут присоединяться к данному ответу. Они определяют выражение исключенных ранее неизвестных задачи Z' через оставшиеся ее неизвестные. Исключенные неизвестные в условиях и посылках текущей задачи Z уже не встречаются. К моменту редактирования ответа кванторная приставка рассматриваемого условия может быть отброшена, а переменные ее добавлены к списку неизвестных задачи. Если эти переменные имели свободные вхождения в утверждения списка B, т.е. совпадали с исключенными неизвестными, то произойдет коллизия - отождествление переменных, обозначающих различные объекты. Для ее предотвращения необходимо заблаговременно переобозначить переменные кванторной приставки так, чтобы они не имели свободных вхождений в B. Это переобозначение выполняется приемом, срабатывающим на уровне 2.

3. Выдача ответа, образованного параметрическим описанием.

   Если условие задачи на описание имеет вид $\exists_{x_1 \ldots x_n} A$ и снабжено комментарием "серия", то рассматриваются все переменные $y_1, \ldots, y_m$ этого условия, являющиеся неизвестными либо встречающиеся в цели (серия $\ldots$). Выделяются все конъюнктивные члены $A_1, \ldots, A_k$ утверждения A, содержащие переменные y_i. Далее проверяются условия:

   (a) Каждое A_j имеет либо вид $y_i = t$, где t известно, либо вид $P(y_i(x) \ldots)$;

   (b) Рассматриваемое кванторное условие не сопровождается комментарием "фильтрсерии";

   (c) Задача не имеет неизвестных, не встречающихся в данном кванторном условии;

   (d) Каждое вхождение неизвестной y вне данного кванторного условия относится к простейшему указателю типа значения этой неизвестной - утверждению $Q(y)$.

   (e) Отсутствует комментарий (контекст $\ldots$) к задаче.

Если все они выполнены, то принимается решение о том, что список условий задачи уже представляет собой почти готовый ответ. К нему применяется стандартная процедура завершающего редактирования (т.е. процедура "редакторответа"), и далее выдается результат ее применения. Уровень срабатывания приема равен 2 либо 3.

Требование отсутствия комментария (контекст ...) объясняется тем, что содержащиеся в нем утверждения, выражающие ранее исключенные неизвестные через неизвестные, определяемые параметрическим описанием, требуют явной подстановки значений согласно этому параметрическому описанию. После подстановки значений могут понадобиться действия по упрощению ответа. В такой ситуации немедленная выдача ответа является преждевременной.

4. Выдача ответа, представляющего собой параметрическое описание числового промежутка.

   В предположениях предыдущего приема, вместо перечисляемых в нем условий, проверяется, что утверждения $A_1, \ldots, A_k$ образуют одно либо два неравенства, определяющих промежуток значений некоторой неизвестной y. Если задача не имеет других неизвестных, а неизвестная y встречается вне рассматриваемого кванторного условия лишь внутри указателя типа ее значения, причем отсутствует комментарий (контекст ...), то предпринимается обращение к процедуре "редакторответа" и выдача ответа. Уровень срабатывания - 2 либо 3.

5. Обращение к вспомогательной задаче на редактирование параметрического описания.

   В предположениях двух предыдущих приемов, оказывается, что приводимые в них условия на утверждения $A_1, \ldots, A_k$ не выполнены. Тогда предпринимается обращение к вспомогательной задаче Z', предназначенной для редактирования рассматриваемого параметрического описания. Предварительно уточняется уровень срабатывания приема: если все утверждения A_i суть неравенства, то он равен 3, иначе - равен 2. Условиями задачи Z' служат все условия текущей задачи Z, отличные от параметрического описания, а также все конъюнктивные члены подкванторного утверждения A. Посылки ее те же, что у текущей задачи. Списки неизвестных и несущественных неизвестных пополняются в задаче Z' переменными $x_1, \ldots, x_n$ кванторной приставки. Если задача Z не имеет цели (серия ...), то вводится цель (серия $x_1 \ldots x_n$), в противном случае старая цель (серия ...) пополняется переменными кванторной приставки. Таким образом, цель (серия ...) перечисляет все переменные задачи, являющиеся параметрами внешних редактируемых параметрических описаний. В задачу Z' переносятся прочие цели задачи Z, а также добавляется цель "учетответа". Последняя служит сигнализатором режима редактирования параметрического описания.

   Если на задачу Z' получен ответ R, отличный от символа "отказ", то на этот ответ навешивается квантор существования по всем входящим в R переменным списка $x_1, \ldots x_n$. Полученное таким образом утверждение упрощается с помощью пакетного нормализатора "нормсуществует", и результат выдается в качестве ответа. В особых случаях предпринимаются дополнительные шаги по исключению из ответа имеющихся в нем кванторов существования.

   Данный прием часто используется при решении тригонометрических уравнений. После получения серии корней $\exists_n(n - \text{целое} \ \& \ x = f(n))$ обычно бывает нужно учесть дополнительные ограничения на x, например, вытекающие из условий на область допустимых значений. При решении указанной выше вспомогательной задачи Z', в которой внешний квантор существования отброшен, появляется возможность подставить выражение для x в эти дополнительные ограничения. Их анализ приводит к отбрасыванию некоторых подсерий найденной серии, и таким образом получается окончательный ответ.

Усмотрение при редактировании ответа параметрического описания для известной переменной, встречающейся в условиях с неизвестной, и подстановка определяемого им значения этой переменной

Пусть при редактировании ответа задачи на описание в списке условий обнаруживается утверждение вида $\exists_{x_1 \dots x_n}(y_1 = t_1 \ \& \ \dots y_m = t_m \ \& \ A)$, не содержащее неизвестных. Пусть также $y_1, \dots, y_m$ - различные переменные, не встречающиеся в выражениях $t_1, \dots, t_m$ и отличные от переменных кванторной приставки. Такая ситуация, в которой появляется параметрическое описание для известного параметров задачи, может возникнуть после решения уравнения относительно этого параметра, а само уравнение - возникнуть при разборе случаев, как указывающее на некоторый вырожденный подслучай. Если известный параметр появляется в условиях, содержащих неизвестные, то делается попытка получить параметрическое описание для неизвестных. Для этого рассматриваются все условия $B_1, \dots, B_k$, содержащие переменные $y_1, \dots, y_m$ и отличные от исходного кванторного условия. Находятся результаты $C_1, \dots, C_k$ подстановки в них выражений $t_1, \dots, t_m$ вместо переменных $y_1, \dots, y_m$, к которым применяется процедура быстрого упрощения. Затем вводится вспомогательная задача на описание Z', условиями которой являются утверждение $\exists_{x_1 \dots x_n}(C_1 \ \& \ \dots \ \& \ C_k \ \& \ y_1 = t_1 \ \& \ \dots \ \& \ y_m = t_m \ \& \ A)$ и все остальные (т.е. не содержащие $y_1, \dots, y_m$) условия текущей задачи. Первое из перечисленных условий снабжается комментарием "серия"; вместо цели "редакция" задаче Z' дается цель "упростить". Это означает, что от редактирования ответа предпринимается откат к повторному рассмаотрению списка условий. Далее задача Z' решается, и найденный на нее ответ (либо отказ) выдается как ответ на текущую задачу.

Переобозначение переменных параметрического описания, определяемое вспомогательным условием

В некоторых случаях при редактировании параметрического описания может оказаться, что целесообразен переход к другим параметрам. Примеры такого рода возникали в связи с редактированием ответов для дифференциальных уравнений. Переход к новым параметрам задается условием вида $\exists_{x_1 \dots x_n}(y_1 = t_1 \ \& \ \dots \ \& \ y_n = t_n \ \& \ A)$, которое заблаговременно вводится дргуим приемом. Это условие снабжается комментарием "заменапеременной". Старые параметры суть $y_1, \dots y_n$; новые - $x_1, \dots, x_n$. Предполагается, что количества этих параметров одинаковы. Утверждение A и выражения $t_1, \dots, t_n$ не содержат старых параметров. Прием, реализующий переход к новым параметрам x_i, использует для их обозначения старые переменные y_i. Он активизируется при сканировании задачи на описание, имеющей цель "учетответа" (см. выше). Проверив наличие комментария "заменапеременной", прием определяет выражения $s_1, \dots, s_n$, получаемые из $t_1, \dots, t_n$ подстановкой переменных $y_1, \dots, y_n$ вместо $x_1, \dots, x_n$. Затем во всех прочих условиях предпринимается подстановка выражений $s_1, \dots, s_n$ вместо $x_1, \dots, x_n$. Такая же подстановка предпринимается в термах комментариев (сопровождение ...). В заключение переобозначения определяющее замену переменных кванторное условие отбрасывается, а утверждение A добавляется к списку условий. Прием срабатывает на уровне 0.

Усмотрение рода объекта, определяемого внутри параметрического описания

Если при редактировании ответа задачи на описание встречается параметрическое описание $\exists_{x_1 \dots x_n}(y = t \ \& \ A)$, где y - неизвестная задачи; t - выражение без неизвестных, причем заголовок выражения t определяет тип P значения этого выражения, то исключается (если оно есть) условие $P(y)$. Прием срабатывает на уровне 2.

Элиминация условного выражения под квантором существования

Если внутри кванторного утверждения $\exists_{x_1 \dots x_n} A$ встречается условное выражение "вариант$(x = p \ t_1 \ t_2)$", причем x - переменная кванторной приставки, а выражение p не имеет вхождений переменных, связанных внутри A, то находятся результаты A_1, A_2 замены в утверждении A всех вхождений данного условного выражения на t_1 и t_2, соответственно. Затем предпринимается упрощение утверждений $\exists_{x_1 \dots x_n}(x = p \ \& \ A_1)$, $\exists_{x_1 \dots x_n}(\neg(x = p) \ \& \ A_2)$, и исходное кванторное утверждение заменяется на их дизъюнкцию. Уровень срабатывания приема равен 3. Прием гарантирует исключение связанной переменной x по крайней мере в первом дизъюнктивном члене.

Переход от условия существования к условию непустоты области определения отображения

Пусть в некотором терме встречаются одновременно утверждение $\exists_{x_1 \dots x_n} A$ и выражение "отображение$(y_1 \dots y_n \ B \ t)$", причем усматривается, что первое совпадает (с точностью до переобозначения связанных переменных и допустимых перестановок операндов) с утверждением $\exists_{y_1 \dots y_n} B$. Тогда предпринимается замена утверждения существования на утверждение "не(равно(класс$(y_1 \dots y_n \ B) \ \emptyset$))". Уровень срабатывания приема равен 4.

1.10 Приемы символа "равно"

Равенство относится к общелогическим символам, так как может встречаться в любой предметной области. Основая часть его приемов реализована на ЛОСе.

Равенство с совпадающими частями

Утверждение $a = a$ заменяется на логическую константу "истина". Уровень срабатывания этого приема равен 0.

Выдача ответа задачи на описание, единственное неизвестное условие которой имеет вид $x = t$

Если задача на описание имеет условие $x = t$, где x - неизвестная, t - выражение, не содержащее x, причем все прочие условия не содержат неизвестных, то обычно решение этой задачи прекращается, и для выдачи ответа предпринимается обращение к процедуре "редакторответа". Уровень срабатывания приема равен 0. Предварительно проверяется ряд требований, из числа которых приведем следующие:

1. Задача не имеет цели "редакция", цели "исследовать" либо цели "функционально". В первом случае ответ уже редактируется; во втором - нет необходимости явно выражать неизвестные через известные; в третьем - применяется особый прием.
2. Задача не имеет цели (известно ...). При такой цели нужно не просто выразить неизвестную через известные, но выразить ее через специальное подмножество известных. Схема решения задачи в этой ситуации - особая, использующая блок анализа.
3. Задача не имеет цели "замещение". Такая цель требует, чтобы неизвестные вообще не входили в ответ.
4. Если задача имеет цель "стандравно", т.е. возникла из задачи на исследование, имеющей цель "известно", то проверяется отсутствие условий, не имеющих свободных переменных. Такие условия эквивалентны логическим константам, и требуется сначала определить их истинность либо ложность. Это важно для усмотрения невыполнимых подслучаев (например, при анализе вариантов построения геометрического чертежа).
5. Задача не имеет цели (обозначение ...), некоторая переменная которой входит в условия. В этой цели перечисляются вспомогательные переменные, появление которых в ответе недопустимо.

Если задача имеет цели "полный", "пример", то, во-первых, проверяется допустимость найденного ответа $x = t$ с точки зрения цели (независит ...), и , во-вторых, предпринимается попытка усмотреть истинность всех известных условий в контексте списка посылок. После этого известные условия отбрасываются, и лишь затем реализуется обращение к процедуре "редакторответа".

Выдача ответа задачи на описание, имеющей цель "функционально"

Если задача имеет цель "функционально", то в ней нужно лишь установить факт однозначного определения значений неизвестных по значениям известных параметров. Явного выражения для неизвестных получать не требуется, причем в процессе решения можно вводить новые параметры, выражаемые однозначно через исходные. Такие задачи возникают в комбинаторике, при замене переменных в параметрическом задании класса объектов. Если в процессе решения для каждой неизвестной x, не являющейся несущественной, обнаруживается условие вида $x = t$, где t известно, то в качестве ответа выдается конъюнкция условий задачи. Уровень срабатывания приема равен 0.

Перестановка левой и правой частей равенства

Равенство $t_1 = t_2$ в посылках или условиях задачи определяет два различных обозначения t_1, t_2 одного и того же объекта. Обычно оно используется для стандартизации таких обозначений - t_1 повсюду в "зоне действия" равенства заменяется на t_2. Исключение составляют случаи, оговоренные специальными комментариями к равенству либо к заменяемому вхождению терма t_1. Однако, целесообразно переходить от "более сложных" в том или ином смысле обозначений к "более простым". Поэтому до применения замены, основанной на равенстве, предпринимается такая перестановка его частей, чтобы слева располагалось "более сложное" выражение. Это происходит при текущем уровне, равном 1. Заметим, что прием замены срабатывает тоже на уровне 1, но программа приема, переставляющего части равенства, расположена до программы приема замены.

Для определения целесообразности перестановки частей равенства служит оператор "ориентацияравенства(x1 x2 x3 x4)". Ему сообщается координата (x2,x3,x4) вхождения равенства в задачу x1. Если перестановка частей необходима, то оператор истинен, иначе - ложен. Перестановку частей равенства - посылки либо условия - блокирует комментарий "ориентацияравенства" к данному терму задачи. В ряде случаев этот комментарий игнорируется (наиболее часто - для некорневых вхождений равенства). Пусть процедуре "ориентацияравенства" переданы координаты вхождения равенства, имеющего части t_1, t_2 (в произвольном порядке). Приоритеты при перестановке этих частей определяются следующим списком ситуаций (каждая очередная рассматривается лишь при условии, что предыдущие не имели места):

1. Если t_1 - неизвестная, а заголовком t_2 служит символ "отображение", причем задача - на исследование и имеет цель "иследовать", то t_1 размещается справа;

2. Если задача - на исследование и имеет цель "исследовать"; t_1 - переменная, t_2 - неоднобуквенный терм, не содержащий символа "отображение" и переменной t_1, то t_1 размещается слева;

3. Если равенство подчинено описателю "класс"; все промежуточные символы между символом этого описателя и равенством - либо конъюнкции, либо кванторы существования, причем t_2 - переменная кванторной приставки описателя, а t_1 - нет, и t_2 не встречается в t_1, то t_2 размещается слева;

4. Пусть задача - на описание и имеет цели (связка . . .), "учетответа", т.е. происходит редактирование параметрического описания для решения системы функциональных (например, дифференциальных уравнений). Если t_1 имеет заголовок "значение" и содержит переменные цели "связка", а t_2 - не содержит таких переменных, то t_1 размещается слева;

5. Если равенство расположено в условиях задачи на описание, либо задача - на исследование, причем t_1 содержит неизвестные, а t_2 - не содержит, то t_1 размещается слева;

6. Если равенство расположено в условиях задачи на описание, либо задача - на исследование и не имеет цели "известно"; обе части равенства содержат неизвестные, причем t_1 - неизвестная и не входит в t_2, то t_1 размещается слева. Если оба терма t_1, t_2 - неизвестные, то перестановки не происходит;

7. Если задача - на исследование и имеет цель "известно", t_1, t_2 - неизвестные этой задачи, причем одна из них является неизвестной внешней задачи на описание, а другая - не является, то последняя размещается слева;

8. Если задача - на исследование и имеет цель "известно", t_2 - неизвестная этой задачи, принимающая численные значения, а t_1 не содержит t_2 и содержит неизвестную, принимающую векторные значения, то t_2 размещается справа;

9. Пусть задача - на исследование и имеет цель "известно"; t_1 - неизвестная, принимающая числовые либо векторные значения и не входящая в терм t_2, который не содержит неизвестных, принимающих не числовые и не векторные значения. Тогда t_1 размещается слева. Если оба терма - неизвестные, перестановка не выполняется;

10. Если задача - на исследование и имеет цель "известно"; t_1 содержит неизвестную, принимающую не числовые и не векторные значения, а t_2 не содержит такой неизвестной, то t_1 размещается слева;

11. Если задача - на исследование и имеет цель "известно"; t_1 принимает числовое либо векторное значение и является либо переменной, либо операцией от операндов, хотя бы один из которых не числовой и не векторный; t_2 - операция от числовых либо векторных операндов, то t_1 размещается слева;

12. Если задача - на доказательство, то для нее предусмотрены аналоги последних семи пунктов. При этом неизвестной задачи считается любой ее параметр, не входящий в комментарий (известно ...);

13. Если задача - на доказательство, t_1 - параметр, по которому ведется индукция и не входит в t_2, то t_1 размещается справа;

14. Если равенство - не корневое; t_1 - переменная, не входящая в t_2 и связанная внешним квантором либо описателем; t_2 - либо не переменная, либо не связано внешними кванторами и описателями, то t_2 размещается справа;

15. Если равенство - корневое; t_1 представляет собой переменную, а t_2 имеет заголовок "отображение" и не содержит t_1, то проверяется наличие посылки либо условия, к контексту которого относится равенство, в которое переменная t_1 входит не под символом "значение" либо область. Если такая посылка (условие) нашлась, то t_1 размещается справа;

16. Если равенство представляет собой посылку задачи на описание, имеющей цель (независит ...), причем t_1 не имеет связанных переменных, а t_2 имеет, то t_1 размещается справа;

17. Если текущая задача не имеет типа "исследовать" либо имеет цель "противоречие", причем t_1 - переменная, а t_2 не является переменной и не содержит t_1, то t_1 размещается слева;

18. Если t_1 содержит переменную, а t_2 не содержит переменных, то t_1 размещается слева;

19. Если t_1 имеет заголовок "значение", а t_2 не имеет такого заголовка и не содержит t_1, то t_1 размещается слева;

20. Если равенство представляет собой консеквент корневой кванторной импликации; текущая задача - на исследование, причем t_1 содержит атомарные числовые выражения, отличные от констант, переменных и условных выражений, а t_2 таковых не содержит, то t_1 размещается слева. Этим достигается приведение кванторной импликации к такому виду, чтобы ее можно было использовать для выражения числовых характеристик нечисловых объектов через числовые параметры;

21. Если t_1 имеет меньшую длину, чем t_2, то t_2 размещается слева;
22. Если терм t_1 лексикографически предшествует терму t_2 (в силу предыдущего пункта, длины этих термов равны), то в случае корневого равенства t_1 размещается справа, иначе - слева.

Заметим, что кроме данного общего приема, перестановка частей равенства выполняется множеством специальных приемов, реализованных на ГЕНОЛОГе. Эти приемы, чтобы закрепить выполненную ими перестановку, сопровождают содержащий равенство терм задачи комментарием "ориентацияравенства". Он отменяет перестановку частей равенства "из общих соображений". Иногда по ходу преобразований фиксация ранее введенной ориентации равенства устаревает, и требуются специальные приемы для ее коррекции - либо путем удаления комментария "ориентацияравенства", либо за счет приемов, игнорирующих этот комментарий и выполняющих перестановку частей равенства с учетом текущей ситуации.

Подстановка значения согласно равенству из текущего контекста

Пусть в задаче обнаружено равенство $t_1 = t_2$, причем текущий уровень равен 1 либо 6. Тогда рассматриваются все вхождения выражения t_1, к контексту которых относится данное равенство, и анализируется целесообразность замены этих вхождений на t_2. Программа приема, выполняющего эти действия, распадается на несколько ветвей, относящихся к различным ситуациям:

1. Вхождение равенства - корневое, т.е. равенство представляет собой некоторую посылку либо условие задачи. Проверяется выполнение нескольких общих условий, при нарушении которых замена нецелесообразна. Прежде всего, устанавливается, что t_1 не входит в t_2 и что текущая задача не является задачей на описание, имеющей цель "известно". Если равенство представляет собой посылку задачи на преобразование, причем t_1 - переменная, не входящая в терм t_2, который имеет достаточно большую длину и не является десятичной константой, то находится число вхождений переменной t_1 в условие задачи. Если это число более 3, то замены блокируются. Если задача не имеет типа "преобразовать", то замены выполняются только на уровне 1, иначе - добавляется попытка на уровне 6. Далее, проверяется отсутствие комментария "блок" к рассматриваемому равенству. Такой комментарий используется для управления данным приемом - вводится заблаговременно другими приемами, если применение равенства для стандартизации обозначений нежелательно. После того, как предварительный анализ целесообразности завершен, рассматриваются два подслучая:

    (a) Заголовок выражения t_1 не является ассоциативным и коммутативным символом. В случае задачи на преобразование вводятся дополнительные сильные ограничения на целесообразность замены: либо t_1 - переменная, не входящая в t_2, либо t_1 имеет заголовок "область", либо t_1 имеет свободные переменные, а t_2 - не имеет. Фактически, прием будет применяться для задач на преобразование лишь в исключительных случаях, так как плохо учитывает целевую специфику задачи.

    Рассматриваются все такие посылки и условия A, к контексту которых относится данное равенство. Для текущего A выполняется цикл проверок целесообразности замены. В основном, они относятся к редко встречающимся специальным случаям. Выделим лишь проверку отсутствия комментария "равно" к A. Такой комментарий используется, если нужно защитить какую-либо посылку либо условие от замен подтермов, выполняемых описываемым приемом. Если множество вхождений t_1 в A непусто, то находится результат B замены всех этих вхождений на t_2. Если этот результат представляет собой равенство - посылку задачи на исследование, имеющей цель "известно", то может быть предпринята попытка его упрощения с помощью пакетных операторов общей стандартизации. Кроме того, может быть выполнена перестановка частей такого равенства. После проверки ряда дополнительных условий целесообразности замены A на B, такая замена выполняется. В особых случаях вместо замены может быть предпринят вывод следствия B при сохраненном A.

По окончании просмотра всех термов A предпринимается коррекция тех комментариев, где нужна синхронная с посылками и условиями замена t_1 на t_2. Особо рассмотрен случай комментария (сопровождение ...).

(b) Заголовок выражения t_1 - коммутативный и ассоциативный символ f. Таким образом, t_1 имеет вид $f(r_1 \dots r_m)$. Этот случай аналогичен предыдущему, однако рассматриваются не явные вхождения терма t_1 в текущий терм A, а вхождения подтермов $f(\dots)$, операнды которых включают все термы $r_1, \dots, r_m$. Замена затрагивает только последние - вместо них вводятся корневые f - операнды терма t_2. Для случая задач на преобразование предусмотрены еще более сильные ограничения - замены предпринимаются только при условии, что t_1 имеет свободные переменные, а t_2 не имеет.

2. Рассматриваемое вхождение равенства $t_1 = t_2$ не является корневым. Тогда уровень срабатывания приема равен 1. Равенство должно являться либо операндом конъюнкции, либо антецедентом кванторной импликации. Предусмотрена проверка нескольких простых и редко нарушаемых условий целесообразности замены. Например, замена блокируется, если равенство относится к стандартному уравнению поверхности. Случай коммутативного и ассоциативного заголовка t_1 особо не выделяется. Просматриваются подчиненные той же конъюнкции или импликации, что и равенство, вхождения подтерма t_1, в контекст которых попадает равенство. Они заменяются на t_2. Как и в предыдущих случаях, корректируется комментарий (сопровождение ...).

Описанный прием стандартизации обозначений, использующий равенство, является чрезвычайно общим и часто применяемым. По мере обучения решателя он сопровождался большим количеством различных эвристических ограничителей. Этот процесс, очевидно, будет продолжаться по мере дальнейшего обучения, причем какие-то из созданных фильтров могут измениться, а какие-то - вообще оказаться исключенными. В некоторых разделах, например, в геометрии, данный прием, возможно, применяется сейчас излишне часто и приводит к образованию неоправданно громоздких выражений. Впрочем, проведенная регулировка решателя пока позволяет справляться с таким недостатком, а сколь-нибудь сильное дополнительное ограничение приема потребует значительной перерегулировки. Попытку такой оптимизации можно будет предпринять впоследствии. На данном этапе работы с решателем происходит лишь предварительное накопление информации, которая позволит создать следующие, более эффективные его версии.

Усмотрение ложности равенства из различия типов его частей

В логической системе предусмотрена некоторая древовидная классификация типов объектов. Если два объекта относятся к различным ветвям этой классификации, то они заведомо различны. Чтобы распознавать одноместные предикаты, задающие тип объекта, служит справочник "родобъекта". Такими предикатами являются: "число(x)", "комплексное(x)", "целое(x)", "точка(x)", "множество(x)", и т.п. Для указания на список всех надтипов данного типа используется справочник "род". Например, на логическом символе "целое" он выдает набор символов "рациональное", "число", "комплексное". Список основных типов объектов пополняется по мере обучения решателя; вся информация о нем заключена в указанных справочниках. Кроме того, введен справочник "тип", указывающий по заголовку операции либо константы набор типов объектов, которые могут являться ее значениями. Иногда здесь приводится полный список, но чаще - только минимальные типы. Например, может быть указан тип "число", но отброшен тип "комплексное". Процедура "родобъекта(...)" позволяет определить набор типов объектов, которые могут являться значениями заданного подвыражения задачи. Она используется специальным приемом, сравнивающим такие два списка для левой и правой частей усмотренного в задаче равенства. Если данные списки не пересекаются, то равенство заменяется на константу "ложь". Уровень срабатывания приема равен 3.

Усмотрение различия константных выражений, являющихся именами

Некоторые константные выражения используются в качестве имен объектов; понятие "имя" при этом предполагает различие объектов, имеющих различные имена. Например, именами являются десятичные записи чисел. Кроме того, предполагается, что однобуквенные выражения, являющиеся логическими символами, также суть имена. Такое соглашение, разумеется, запрещает вводить различные логические символы для обозначения одного и того же объекта. Вообще, всегда считается, что значением однобуквенного выражения, образованного логическим символом, служит сам этот символ, с которым и отождествляется обозначаемый им объект предметной области. На уровне 0 срабатывает прием, распознающий в левой и правой частях равенства два различных имени. В такой ситуации он заменяет равенство на константу "ложь".

Исключение посылки задачи на доказательство, имеющей вид равенства переменной, не встречающейся в других посылках и условии

Если посылка задачи на доказательство имеет вид $a = t$, где a - переменная, не встречающаяся в выражении t, а также в прочих посылках, условии и в комментариях (известно ...), (неизвестные ...), то эта посылка удаляется. Уровень срабатывания приема равен 1.

Применение тождества из посылок для исключения переменной в условии задачи на преобразование

Пусть посылка задачи на преобразование имеет вид $t_1 = t_2$, причем замена всех вхождений выражения t_1 в условие задачи на выражение t_2 позволяет избавиться в этом условии от некоторой переменной x, не добавляя новых переменных. Если задача не имеет других целей, кроме цели "упростить", то указанная замена реализуется. Уровень срабатывания приема равен 5.

Исключение неизвестной задачи на описание, явно выраженной через остальные неизвестные

Если в условиях задачи на описание возникает равенство $x = t$, где x - неизвестная, t - выражение, не содержащее этой неизвестной, то обычно предпринимается переход к решению новой задачи, не содержащей x. Уровень срабатывания приема, выполняющего такой переход, равен 1.

Если неизвестная x является несущественной, прочие условия задачи не содержат x, и нет комментария (контекст ...), в котором упоминается x, то происходит отбрасывание условия $x = t$, и работа приема на этом завершается. Предварительно проверяется отсутствие цели (независит ...), запрещающей неизвестной x зависеть от какой-либо переменной выражения t.

Если указанная ситуация не имеет места, то проверяется, что задача не имеет целей (известно ...), "редакция" и число ее неизвестных не менее 2. Если цель (независит ...) запрещает неизвестной x зависеть в ответе задачи от какой-либо свободной переменной выражения t, то прием блокируется. В противном случае создается вспомогательная задача на описание Z'. Ее посылки - те же, что у текущей задачи Z, но дополнительно вводится фиктивная посылка - "фиктпосылка(x)". Она нужна, чтобы процедуры, которые при последующем решении будут выбирать новые переменные, знали, что переменная x уже занята. Хотя эта переменная исключается из условий решаемых далее задач, но на этапе завершающего редактирования ответа равенство $x = t$ будет извлечено из комментария (контекст ...), и переменная x вновь появится. Условиями задачи Z' являются результаты подстановки выражения t вместо переменной x во все остальные условия задачи Z. Комментарии к задаче Z' переносятся из Z с помощью справочника "редукция". Цели корректируются соответственно исключению неизвестной x - она удаляется из списков неизвестных и несущественных неизвестных. Для измененных условий задачи Z' корректируются комментарии (сопровождение ...). Если неизвестная x не являлась несущественной, то все несущественные неизвестные выражения t рассматриваются в задаче Z' как существенные. В задачу Z' переносятся из задачи Z комментарии "разборслучаев" к дизъюнктивным условиям.

Для сохранения информации об отброшенном условии $x = t$ служит комментарий (контекст A B). У него B - список всех отброшенных условий, возникавших на различных этапах решения; A - задача, цели которой следует учитывать при завершающем редактировании ответа, после добавления к нему утверждений списка B. Прием, выдающий ответ на тот или иной подслучай задачи Z', обратится к процедуре "редакторответа", осуществляющей учет комментария (контекст ...). Она пополнит ответ утверждениями списка B, подставит в добавленные утверждения найденные значения неизвестных и упростит результат. Для этого будет решаться еще одна вспомогательная задача на описание, снабженная целью "редакция".

После регистрации условия $x = t$ в комментарии (контекст ...) предпринимается обращение к решению задачи Z'. Для полученного ответа R проверяется выполнение ограничений, накладываемых на x целью (независит ...). Затем корректируются комментарии (выводимо ...),(сопровождение ...), и выдается ответ R.

Выдача ответа задачи на описание, если все ее условия суть равенства, определяющие численные значения неизвестных

Если все условия задачи на описание суть либо равенства $x = t$, где x - неизвестная, t - число, либо утверждения "число(x)", то предусмотрена ускоренная выдача в качестве ответа конъюнкции условий. Предварительно проверяется отсутствие комментария (контекст ...), накапливающего ранее найденные фрагменты ответа. Проверяется также, что для каждой неизвестной задачи имеется равенство, определяющее ее значение. Уровень срабатывания приема равен 0.

Равенство в условиях задачи на описание, явно выражающее один из параметров через другие параметры

Если условием задачи на описание является не содержащее неизвестных равенство $a = t$, где a - переменная, не входящая в выражение t, то предпринимается попытка перейти к решению вспомогательной задачи Z', полученной из текущей задачи подстановкой выражения t вместо a как в посылках, так и в условиях. Предварительно проверяется, что задача не имела цели (известно ...), а указанное равенство возникло при разборе случаев как фрагмент одной из альтернатив. Перед решением задачи Z' предпринимается упрощение максимальных подвыражений ее посылок и условий, изменившихся после подстановки. После получения на задачу Z' отличного от символа "отказ" ответа R формируется конъюнкция $x = t \ \& \ R$, которая и выдается как ответ на текущую задачу. Уровень срабатывания приема равен 1.

Подбор примера объекта, отличного от заданных объектов

Пусть все условия задачи на описание суть $x \neq t_1, \ldots, x \neq t_n$, где x - единственная неизвестная задачи; $t_1, \ldots, t_n$ - выражения без неизвестных. Пусть также задача имеет цель "пример". Если все выражения $t_1, \ldots, t_n$ суть имена, то находится некоторое число a, отличное от всех этих имен, и рассматривается равенство A вида $x = a$. Иначе в качестве A берется равенство "$x =$ внешнийэлемент($\{t_1, \ldots, t_n\}$)". Если задача не имела комментария (контекст ...), содержащего ранее найденные фрагменты ответа, то A выдается как ответ. Иначе все условия текущей задачи заменяются на единственное условие A, и для получения ответа предпринимается обращение к процедуре "редакторответа". Уровень срабатывания приема равен 2.

Вывод в посылках из кванторной импликации, антецедент которой имеет вид равенства, путем унификации частей этого равенства

Пусть в посылках текущей задачи встречается кванторная импликация вида $\forall_{x_1 \ldots x_n}$ $(t_1 = t_2 \to A)$ либо вида $\forall_{x_1 \ldots x_n}(B \to \neg(t_1 = t_2))$, где A, B - элементарные утверждения; выражение t_1 не содержит переменных кванторной приставки, а выражение t_2 содержит все эти переменные. Если текущая задача - на описание и имеет цель "пример", либо на доказательство, то предпринимается попытка подобрать такие выражения $s_1, \ldots, s_n$, подстановка которых в t_2 вместо переменных $x_1, \ldots, x_n$ дает (с точностью до простейших тождественных преобразований) выражение t_1. Если это удается, то находится результат C подстановки в A (соответственно, в $\neg B$) выражений $s_1, \ldots, s_n$ вместо $x_1, \ldots, x_n$, который выводится в качестве следствия. Уровень срабатывания приема равен 2.

Решение относительно параметра числового равенства без неизвестных

Если в условиях задачи на описание встречается числовое равенство $t_1 = t_2$, не содержащее неизвестных и не являющееся выражением переменной через другие переменные, то предпринимается попытка разрешить его явным образом относительно какого - либо встречающегося в нем атомарного числового параметра - переменной или числовой характеристики нечисловых объектов. Такая же попытка предпринимается для условия вида $\neg(t_1 = t_2)$, не содержащего неизвестных, если только оно имеет единственную свободную переменную. Предварительно проверяется ряд условий на целевую установку задачи: отсутствие целей (известно ...), "замещение", "стандравно" и др. Проверяется также, что рассматриваемое утверждение не используется для сопровождения по о.д.з. Выбор атомарного числового параметра и обращение к вспомогательной задаче для разрешения относительно него выполняются процедурой "определениепараметра". Процедура ищет такой параметр, для которого уравнение будет иметь самый простой вид. Если после решения уравнения получается дизъюнкция, содержащая в своих подслучаях явные выражения для какого-либо известного параметра, встречающегося в условиях с неизвестными, то эта дизъюнкция сопровождается комментарием "разборслучаев". Уровень срабатывания приема равен 3.

Решение относительно числового параметра равенства, содержащегося в списке посылок

Если числовое равенство $t_1 = t_2$ встречается в списке посылок, то может быть применена процедура разрешения его относительно некоторого числового атома, аналогичная описанной в предыдущем пункте. Предварительно проверяется ряд условий на целевую установку задачи, из которых выделим следующие:

1. Если задача - на исследование, то она имеет цель "противоречие";
2. Если задача - на описание, то она не имеет целей "редуцирование", "замещение", (известно ...).
3. Если задача - на преобразование и не имеет целей, кроме цели "упростить", то все переменные равенства входят в ее условие;
4. Если задача - на преобразование, то не имеет целей "известны", "частнпроизв", "длина";

Проверяется, далее, что в равенство не входят описатели "класс", "отображение" и что оно не является явным выражением одной переменной через другие. Для выбора числового атома и разрешения равенства применяется та же процедура "определениепараметра", что и в предыдущем приеме. Уровни срабатывания приема - 1, 3, 5. На каждом из них устанавливается свой (равный, соответственно, 2, 5 и 20) максимальный уровень вспомогательной задачи, обеспечивающей разрешение. В приеме особо рассматривается случай комплекснозначного равенства, где применяется аналогичная процедура "Определениепараметра".

Возвращение к условиям задачи на описание после того, как анализ объединенного списка условий и посылок привел к явному выражению для одной из неизвестных

Если при решении задачи на исследование Z', являющейся блоком анализа задачи на описание Z, получается равенство вида $x = t$, где x - неизвестная задачи Z; t - выражение, не содержащее x, то может быть предпринят обрыв решения задачи Z' и возвращение к решению задачи Z, с перенесением в нее найденного равенства.

Это делается лишь в тех случаях, когда задача Z' не имеет целей "известно", "контрольвывода", "исследовать". Кроме равенства $x = t$, в задачу Z переносится ряд сопутствующих утверждений, и из условий задачи Z исключаются те утверждения, которые являются следствиями перенесенных в нее новых утверждений. Это делает процедура "замещениеусловий". Уровень срабатывания приема равен 0. Как видно из приведенных выше ограничений, для задач с целью "известно" прием заблокирован. Это объясняется тем, что здесь нецелесообразно спешить с передачей во внешнюю задачу части ответа - экономнее сначала найти весь ответ в рамках задачи на исследование. Поэтому введен специальный прием, анализирующий равенства $x = t$ для задач на исследование, имеющих цель "известно" (см. ниже). При усмотрении полного ответа он обеспечивает его упрощение и выдачу.

Исключение внешней фиктивной операции для равенства в консеквенте кванторной импликации

Пусть кванторная импликация имеет вид $\forall_{x_1 \ldots x_n}(A_1 \ \& \ \ldots \ \& \ A_m \rightarrow f(x_i t_1) = f(x_i t_2)$, где переменная x_i не встречается в t_1, t_2 и в существенных посылках A_j. Тогда проверяется, что операция f имеет единицу a по переменной x_i; эта единица подставляется во все антецеденты, содержащие x_i, и проверяется, что они являются следствиями прочих антецедентов. После этого выполняется замена кванторной импликации на утверждение $\forall_{x_1 \ldots x_{i-1} x_{i+1} \ldots x_n}(A_{j_1} \ \& \ \ldots \ \& \ A_{j_k} \rightarrow t_1 = t_2)$, где $A_{j_1}, \ldots, A_{j_k}$ - все не содержащие x_i антецеденты. Уровень срабатывания приема равен 1.

Попытка решения уравнения из блока анализа, имеющего единственную неизвестную

Если вывод следствий в блоке анализа Z' некоторой задачи на описание Z дает уравнение $t_1 = t_2$, содержащее единственную числовую неизвестную x, то может быть предпринято обращение к вспомогательной задаче на описание для разрешения данного уравнения относительно x. Уровни, на которых предпринимается эта попытка, равны 1, 3 и 5. При наличии в уравнении неизвестных тригонометрических подвыражений допускаются только уровень 5, либо, для не очень длинных уравнений, уровень 3. Далее проверяется выполнение следующих требований:

1. Уравнение не дает явного выражения для x;
2. Из контекста усматривается, что x - числовая переменная;
3. Не существует дизъюнктивно-конъюнктивной посылки, фрагмент которой дает явное выражение для x;
4. Рассматриваемое уравнение не было перенесено в блок анализа из списка условий внешней задачи на описание, имеющей единственную неизвестную и не имеющей цели (известно ...);
5. Не существует более короткого уравнения, удовлетворяющего тем же условиям, для которого еще не выполнялась попытка применения данного приема;

Тогда создается вспомогательная задача на описание Z'', посылками которой служат все не содержащие неизвестных посылки задачи Z', а единственным условием - рассматриваемое уравнение. Цели этой задачи суть "полный", "прямойответ", "или", "явное", (неизвестные x). Если задача Z' имеет цель "известно", то к данному списку добавляются также цели "упростить" и "стандравно". Заметим, что последняя цель используется во всех вспомогательных задачах на описание, образованных подсистемой уравнений задачи на исследование Z', имеющей цель "известно". Она указывает на возможность применения несколько упрощенной процедуры решения, учитывающей потребности задачи Z'.

Условия и посылки задачи Z'' обычно бывает целесообразно пополнить некоторыми дополнительными утверждениями, извлекаемыми из списка посылок блока анализа. Для такого пополнения в процессе обучения решателя были отобраны правила, перечисляемые ниже. Заметим, что занесение в Z'' слишком сложных сопровождающих утверждений, ограничивающих допустимые значения неизвестной x, оказалось нецелесообразным - во многих случаях оно сильно замедляло решение задачи. Столь же нежелательным и по той же причине оказался и недобор сопровождающих утверждений. Поэтому понадобилась эмпирическая оптимизация, после которой возник приводимый ниже перечень правил. Они формируют объединенный список S дополнительных условий и посылок:

1. В S включаются все утверждения, необходимые для сопровождения уравнения $t_1 = t_2$ по о.д.з. и имеющие свободные переменные;

2. Если имеется посылка "целое(x)", то она включаестя в S;

3. Если есть основания предположить, что при решении задачи будет полезна информация о неотрицательности x (например, если x - значение заведомо неотрицательной числовой характеристики нечислового объекта, либо расположено под достаточно "сложной" операцией), то выполняется быстрая проверка неотрицательности x. В случае успеха список S пополняется утверждением $0 \leq x$.

4. Если блок анализа имеет посылку вида ”расстояние(AB)= r”; r - числовое выражение, все параметры которого встречаются в задаче Z'', то в S заносится неравенство $0 \leq r$, в правой части которого отбрасываются заведомо положительные множители;

5. Если уравнение содержит неизвестную x под тригонометрической функцией, то просматриваются посылки блока анализа, имеющие вид "угол(...)= r" либо "уголмежду(...)= r". В S заносится неравенство $0 \leq r$ или, если быстро удается усмотреть строгую положительность, неравенство $0 < r$. Если быстро удается усмотреть неравенство $0 \leq \pi - 2r$, то оно также заносится в S. Иначе - в S заносится $0 \leq \pi - r$.

6. Пусть в блоке анализа имеется равенство $T = r$, где r - либо переменная, либо известное выражение малой длины (например, не более 4). Пусть также T имеет своим заголовком один из символов "угол", "уголмежду", "расстояние", "площадь". Тогда в S заносится неравенство $0 \leq r$. Если T имеет вид "расстояние(AB)", причем удается усмотреть различие точек A, B, то данное неравенство заменяется на строгое. Если r - переменная, то далее рассматриваются следующие подслучаи:

   (a) Если заголовок T - символ "угол" либо "уголмежду", то предпринимается попытка усмотреть, что данный угол - острый, т огда в S заносится неравенство $0 < \pi - 2r$. При неудаче предпринимается попытка доказать, что угол меньше пи, и тогда заносится утверждение $0 < \pi - r$. В прочих случаях заносится утверждение $0 \leq \pi - r$;

(b) Если заголовок T - символ "площадь", то предпринимается попытка установить ее положительность. При удаче в S заносится утверждение $0 < r$;

7. Если в блоке анализа имеется посылка $r = \sin T$, где r - переменная, встречающаяся в Z''; T - выражение с заголовком "угол" либо "уголмежду", то в S заносится неравенство $0 \leq r$;

8. Если в блоке анализа имеется посылка $r = T$, где r - переменная, встречающаяся в уравнении $t_1 = t_2$ в четной степени, то предпринимается попытка быстрого усмотрения неотрицательности выражения T. При удаче в S заносится неравенство $0 \leq r$.

9. Если неизвестная x встречается в уравнении внутри тригонометрического подвыражения, причем блок анализа имеет своей посылкой строгое либо нестрогое неравенство с $\pi/2-x$ в одной части и 0 - в другой, то это неравенство заносится в S.

Далее предпринимается упрощение списка S - каждое входящее в него утверждение обрабатывается нормализатором "нормодз" в контексте прочих утверждений списка, а также посылок задачи Z''. Нормализатор "нормодз" уже встречался ранее - он использовался процедурой "одз" для первичной обработки сопровождающих утверждений.

Все содержащие неизвестную x утверждения списка S добавляются к условиям задачи Z'', а остальные - к ее посылкам. После этого предпринимается решение задачи Z'' с максимальным уровнем, равным 6. Если при решении получен "отказ"либо утверждение, отличное от константы "ложь" и не содержащее символа "равно" в каждом из своих дизъюнктивных членов, то уравнение $t_1 = t_2$ помечается комментарием "неизвестная", блокирующим повторные попытки применения приема. В противном случае конъюнктивные члены ответа замещают в задаче Z' рассматриваемое уравнение. Корректируются комментарии "выводимо", "условие", "прообраз".

Выдача полученного в блоке анализа ответа задачи на описание, имеющей цель "известно"

Пусть решается задача на исследование Z' с целью "известно", являющаяся блоком анализа задачи на описание Z, и в ней усмотрено равенство $x = t$, где t - выражение без неизвестных; x - неизвестная задачи Z. Тогда на уровнях 0, 3, 6 предпринимается попытка усмотреть в посылках задачи Z' ответ задачи Z. Прежде всего, проверяется отсутствие в t переменных, выделенных комментарием (вспомпараметр ...). Этот комментарий накапливает переменные, которые при решении задачи Z' вводились как вспомогательные "известные". Они не должны входить в ответ; после того, как значение такой переменной выражается через исходные "известные", она переносится в список неизвестных задачи Z'.

Затем проверяется, что для каждой неизвестной y задачи Z, не выделенной комментарием (вспомнеизвестная ...), среди посылок задачи Z' имеется равенство вида $y = r$, где r не содержит неизвестных и переменных, выделенных комментарием (вспомпараметр ...). Заметим, что комментарий (вспомнеизвестная ...) перечисляет вспомогательные неизвестные, вводившиеся при решении задачи Z'. Они одновременно регистрировались в задаче Z, однако в ответ задачи Z информация о их значениях включаться не будет. Эти неизвестные нужны были лишь для управления ходом решения задачи Z'.

Если задача Z' имеет цель "контроль", то проверяется отличие текущего уровня от 0. Данная цель указывает на то, что рассматривается некоторый подслучай исходной задачи. Вообще говоря, он может оказаться нереализуемым. Например, так часто бывает в планиметрических задачах. Чтобы усмотреть противоречие и отсечь подслучай, предусмотрена пауза перед выдачей ответа. На уровнях от 0 до 2 срабатывают приемы, усматривающие противоречивость посылок задачи Z'. Все они активизируются лишь при наличии цели "контроль".

Наконец, просматриваются всевозможные равенства $y = r$ указанного выше вида, и для каждого из них выполняется упрощение выражения r. Здесь последовательно решаются две задачи на преобразование. Первая из них, имеющая цели "упростить", "известны", предпринимает стандартные действия по упрощению. Уровень ее, как и следующей задачи, равен 4. Вторая задача, с целями "длина" и "учетрезультата", ориентирована на остаточные упрощения и сокращенную переформулировку результата. Цели "известны", "учетрезультата" указывают на данный прием выдачи ответа. В других задачах они не встречаются. После упрощений равенства замещают все ранее имевшиеся условия внешней задачи Z. Чтобы по возвращении к задаче Z сразу же был выдан ответ на нее, данная задача снабжается комментарием "ответ". В заключение прием обрывает решение задачи Z'. Таким образом, никакого дополнительного редактирования ответа в рамках внешней задачи Z уже не происходит - там немедленно выдается конъюнкция равенств, определяющих значения неизвестных.

Усмотрение и решение подсистем числовых уравнений в задаче на исследование, имеющей цель "известно"

Задачи на описание, имеющие цель "известно" (например, планиметрические задачи на вычисление), решаются путем вывода следствий в своем блоке анализа. Если на некоторый момент вывод следствий дает подсистему численных уравнений, в которой количество неизвестных равно количеству уравнений, то может быть предпринята попытка решить ее с помощью вспомогательной задачи. Имеется несколько различных приемов, реализующих данную попытку:

1. Выделение системы из двух уравнений с двумя неизвестными. Прием срабатывает на уровне 6. Его применение начинается с усмотрения уравнения $t_1 = t_2$, содержащего ровно две неизвестных x, y рассматриваемого блока анализа Z' задачи на описание Z, которые обе оказываются числовыми. Составляется список S всех уравнений, не содержащих отличных от x, y неизвестных блока анализа и не являющихся равенствами, выражающими одну неизвестную через другую. Проверяется, что S имеет не менее двух и не более трех элементов. Предпринимается переупорядочение уравнений списка S по возрастанию их длин. Для разрешения относительно неизвестных последовательно выбираются пары уравнений, одно из которых - текущее уравнение $t_1 = t_2$, а другое - отличный от него элемент списка S. Таким образом, в первую очередь рассматриваются более короткие дополнительные уравнения. С помощью комментария (неизвестные ...) блокируются повторные попытки разрешения ранее рассматривавшейся системы. Вводятся накопители P и U, соответственно, посылок и

условий вспомогательной задачи на описание Z''. В программе они обозначены переменными x14 и x12. В список P заносятся все не содержащие неизвестных посылки задачи Z', имеющие общий параметр с отобранными уравнениями. В список U заносятся: отобранная для решения пара уравнений; утверждения "число(x)" и "число(y)"; все неравенства и отрицания равенств, не содержащие неизвестных задачи Z', отличных от x, y; посылки, указывающие на целочисленность значений неизвестных x, y (если таковые имеются). Далее просматриваются всевозможные переменные z, совпадающие с одной из неизвестных x, y либо представляющие собой числовой параметр утверждения списка P. Для них предпринимается пополнение списков P, U перечисляемыми ниже дополнительными утверждениями. Если z - параметр утверждения списка P, то утверждение заносится в список P, иначе - в список U.

(a) Если имеется посылка $z = r$, где заголовок выражения r - один из символов "угол", "расстояние", "площадь", то добавляется утверждение $0 \leq z$, где по мере возможности вместо нестрогого неравенства берется строгое. В случае угла предпринимаются попытки усмотреть и добавить неравенство, указывающее, что угол острый. Кроме того, вводится неравенство, указывающее, что угол не превосходит пи; по мере возможности - строгое.

(b) Если r - синус угла, то добавляется неравенство $0 \leq z$;

(c) Если удается усмотреть, что z неотрицательно либо положительно, то добавляется соответствующее неравенство.

После этого формируется вспомогательная задача на описание Z''. Она имеет цели "полный", "явное", "прямойответ", "одз", "упростить", "стандравно", (неизвестные xy). Если какая - либо из неизвестных x, y была в задаче Z' выделена комментарием (вспомнеизвестная ...), то она регистрируется как несущественная неизвестная задачи Z''.

Задача Z'' решается с максимальным уровнем 6. Если на нее получен ответ R, отличный от символа "отказ", то проверяется наличие в R равенства, дающего явное выражение для какой-либо из неизвестных x, y. Если R содержит дизъюнкцию, то проверяется, что равенство указанного вида имеется либо в каждом ее дизъюнктивном члене, либо в той части R, которая относится к контексту дизъюнкции. Затем из списка посылок блока анализа Z' исключаются те уравнения, которые разрешались, и вместо них заносится утверждение R.

2. Решение системы из трех уравнений с тремя неизвестными. Уровень срабатывания приема равен 5. Он аналогичен предыдущему приему, однако возникает небольшое отличие при определении списка неизвестных. Если текущее уравнение $t_1 = t_2$ имеет ровно три числовых неизвестных x, y, z, то по ним составляется список S всех уравнений блока анализа, не содержащих других неизвестных, кроме x, y, z. Проверяется, что этот список имеет не менее 3 и не более 5 элементов, и далее выполняются те же действия, что в предыдущем приеме. Если текущее уравнение имеет только две числовых неизвестных x, y, то для определения третьей неизвестной z находится еще одно уравнение, имеющее ровно две числовых неизвестных - например, x и z. Затем по тройке неизвестных составляется список S, и далее - как для двух неизвестных.

3. Решение системы из двух уравнений с тремя неизвестными. Пусть блок анализа имеет всего два уравнения U_1, U_2 для каких-то трех своих числовых неизвестных x, y, z, однако есть уравнение $v = t$, выражающее еще одну числовую неизвестную v внешней задачи на описание через x, y. Если можно предположить, что v выразимо через отношение x к y (например, после усмотрения в t дроби, числитель и знаменатель которой содержат неизвестные x, y), то вводится вспомогательная задача для решения системы $U_1, U_2, v = t$ относительно неизвестных y, z, v. При этом переменная x считается известной, так что все посылки с x, не содержащие других неизвестных блока анализа, регистрируются в посылках вспомогательной задачи. В остальном действия аналогичны случаю двух неизвестных. Уровень срабатывания приема равен 5.

4. Решение системы из четырех уравнений с четырьмя числовыми неизвестными. Уровень срабатывания приема равен 7. Если текущее уравнение $t_1 = t_2$ имеет ровно 4 числовых неизвестных x, y, z, v, то составляется список S всех уравнений блока анализа, неизвестные которых содержатся среди x, y, z, v. Проверяется, что этот список имеет ровно 4 элемента, и далее выполняются действия, аналогичные случаю двух неизвестных.

Преобразование вспомогательных параметров блока анализа, через которые выражена неизвестная внешней задачи на описание, во вспомогательные неизвестные

При решении задачи на исследование, имеющей цель "известно", могут возникать вспомогательные переменные, характеризующие те или иные объекты задачи (например, высота треугольника) и которые временно считаются известными. Такие переменные x помечаются комментариями (вспомпараметр x) к посылкам задачи. Равенства, определяющие их значения, временно присоединяются к списку посылок внешней задачи на описание Z. После того, как через вспомогательные параметры удалось выразить какую-либо неизвестную внешней задачи, либо выразить один из них через другие, происходит изменение статуса параметра - он объявляется вспомогательной неизвестной. Прием, выполняющий это преобразование, срабатывает на уровне 0. Он инициируется посылкой вида $y = t$, где y - либо неизвестная задачи Z, либо один из вспомогательных параметров; t - выражение, содержащее вспомогательные параметры. Прием составляет список S всех вспомогательных параметров, встречающихся в данной посылке. В задаче Z он исключает все посылки задачи Z, имеющие переменную списка S, и вводит для переменных v из S комментарии (вспомнеизвестная v). В задаче на исследование Z' - блоке анализа задачи Z - прием заменяет комментарии (вспомпараметр v), где $v \in S$, на комментарии (вспомнеизвестная v). В обоих задачах список неизвестных пополняется переменными из S.

Пополнение списка неизвестных задачи на описание параметрами числовых уравнений

Если количество числовых уравнений задачи на описание больше числа неизвестных, имеющихся в этих уравнениях, однако равно числу всех встречающихся в уравнениях числовых переменных - как известных, так и неизвестных, то может помочь прием, решающий данную систему уравнений в предположении, что все ее переменные - неизвестные. Уровень срабатывания приема равен 9. Он вводит вспомогательную задачу Z', полученную из текущей задачи Z объявлением неизвестными всех переменных S, содержащихся в числовых уравнениях, и перенесением в список условий всех посылок, зависящих от S.

1.11 Приемы символа "эквивалентно"

Символ "эквивалентно" встречается в задачах редко, главным образом в консеквентах кванторных импликаций. Для него понадобились лишь простейшие приемы, выполняющие общую стандартизацию. Приводимый ниже список не претендует на полноту - он лишь показывает, какие именно связанные с эквивалентностью преобразования оказались фактически востребованы.

Совпадающие операнды

Утверждение $A \leftrightarrow A$ заменяется на логическую константу "истина". Уровень срабатывания равен 0.

Противоположные операнды

Утверждение $A \leftrightarrow \neg A$ заменяется на логическую константу "ложь". Уровень срабатывания 0.

Заголовок каждого операнда - отрицание

Утверждение $\neg A \leftrightarrow \neg B$ заменяется на $A \leftrightarrow B$. Уровень срабатывания 0.

Заголовок одного операнда - отрицание, а другой операнд есть дизъюнкция либо конъюнкция отрицаний

Утверждение $\neg A \leftrightarrow (\neg B_1 \vee \ldots \vee \neg B_n)$ заменяется на $A \leftrightarrow (B_1 \,\&\, \ldots \,\&\, B_n)$. Двойственное преобразование применяется, если правая часть представляет собой конъюнкцию отрицаний. Уровень срабатывания 0.

Замена кванторной эквивалентности в условиях задачи на доказательство на две кванторных импликации

Если утверждение $\forall_{x_1 \ldots x_n}(A \rightarrow (B \leftrightarrow C))$ входит в условие задачи на доказательство, то оно заменяется на конъюнкцию утверждений $\forall_{x_1 \ldots x_n}(A \,\&\, B \rightarrow C)$, $\forall_{x_1 \ldots x_n}(A \,\&\, C \rightarrow B)$. Уровень срабатывания приема равен 0.

Замена кванторной эквивалентности на импликацию

В специальных случаях можно заменить кванторную эквивалентность на импликацию. Так, утверждение $\forall_{x_1 \ldots x_n}(C \rightarrow (A \vee B \leftrightarrow A))$ заменяется на $\forall_{x_1 \ldots x_n}(C \,\&\, B \rightarrow A))$. Утверждение $\forall_{x_1 \ldots x_n}(C \rightarrow (A \,\&\, B \leftrightarrow A))$ заменяется на $\forall_{x_1 \ldots x_n}(C \,\&\, A \rightarrow B))$. Уровень срабатывания приемов равен 0.

Выражение бескванторной эквивалентности через дизъюнкцию, конъюнкцию и отрицание

Если утверждение $A \leftrightarrow B$ не расположено под квантором либо описателем и не является условием задачи на доказательство, то оно заменяется на $(A \ \& \ B) \vee (\neg A \ \& \ \neg B)$. Уровень срабатывания равен 1.

Равенство в одной из частей эквивалентности, дающее явное выражение для переменной

Если одна из частей эквивалентности имеет вид $x = t \ \& \ A(x)$, где x - переменная; t - отличное от переменной выражение, не содержащее x, то эта часть заменяется на $x = t \ \& \ A(t)$. Уровень срабатывания равен 2.

Различающиеся фрагменты частей эквивалентности имеют вид отрицаний

Если эквивалентность имеет вид $(\neg A \ \& \ B) \leftrightarrow (\neg C \ \& \ B)$, то она преобразуется к виду $(A \ \& \ B) \leftrightarrow (C \ \& \ B)$. Уровень срабатывания равен 2.

Перенесение в антецедент общего фрагмента двух частей кванторной эквивалентности

Кванторная эквивалентность вида $\forall_{x_1 \dots x_n}(A_1 \ \& \ \dots \ \& \ A_n \rightarrow (B \ \& \ C \leftrightarrow B \ \& \ D))$ преобразуется к виду $\forall_{x_1 \dots x_n}(A_1 \ \& \ \dots \ \& \ A_n \ \& \ B \rightarrow (C \leftrightarrow D))$. В условии задачи на описание, имеющей цель "экв", прием применяется на уровне 0; иначе - на уровне 5.

Кванторная эквивалентность двух равенств с общей переменной

Если кванторная эквивалентность имеет консеквент вида $x = t \leftrightarrow x = s$, где x - переменная кванторной приставки, а выражения s, t не содержат x, то в ее антецедентах предпринимается подстановка t вместо x, консеквент заменяется на $s = t$, а x исключается из кванторной приставки. Уровень срабатывания приема равен 2.

Кванторная эквивалентность допускает разбиение по независимым группам переменных

Если кванторная эквивалентность не имеет антецедентов, а кванторная приставка ее может быть разбита на две подгруппы переменных x и y так, что консеквент имеет вид $(P(x) \ \& \ Q(y)) \leftrightarrow (R(x) \ \& \ S(y))$, то она заменяется на дизъюнкцию утверждений $(\forall_x(\neg P(x)) \vee \forall_y(\neg Q(y))) \ \& \ (\forall_x(\neg R(x)) \vee \forall_y(\neg S(y)))$, $\forall_x(P(x) \leftrightarrow R(x)) \ \& \ \forall_y(Q(y) \leftrightarrow S(y))$. Уровень срабатывания приема равен 4.

Усиление контекста описателя "отображения", встречающегося в кванторной эквивалентности

Пусть кванторная эквивалентность имеет вид $\forall_{x_1 \dots x_n}(A_1 \ \& \ \dots \ \& \ A_n \rightarrow (B \leftrightarrow C \ \& \ D))$; утверждение B содержит описатель "отображение($z \ P \ t$)", причем все свободные переменные конъюнктивного члена C правой части эквивалентности входят в P, а любой другой конъюнктивный член этому условию не удовлетворяет. Тогда эквивалентность преобразуется в конъюнкцию утверждений $\forall_{x_1 \dots x_n}(A_1 \ \& \ \dots \ \& \ A_n \ \& \ C \rightarrow$

$(B \leftrightarrow D))$, $\forall_{x_1 \ldots x_n}(A_1 \ \& \ \ldots \ \& \ A_n \ \& \ B \rightarrow C)$. Первое из них позволяет использовать при преобразованиях, затрагивающих описатель, дополнительные условия C. Уровень срабатывания приема равен 4.

Доказательство эквивалентности путем рассмотрения двух импликаций

Если условие задачи на доказательство Z имеет вид $A \leftrightarrow B$, то на текущем уровне 1 предпринимается последовательное решение двух вспомогательных задач. Первая из них получается добавлением к посылкам задачи Z утверждения A и заменой условия на B; вторая - добавлением к посылкам B и заменой условия на A. Если обе решены, выдается ответ "истина".

1.12 Приемы символа "вариант"

Большая часть общих приемов символа "вариант" реализована на ГЕНОЛОГе и будет рассмотрена в следующих разделах. На ЛОСе реализованы лишь несколько приемов.

Вынесение из-под условного выражения символа одноместной операции

Если условное выражение имеет вид "вариант$(A\ f(t_1)\ f(t_2))$", где f - символ одноместной операции, то оно преобразуется к виду f(вариант$(A\ t_1\ t_2)$). Уровень срабатывания этого приема равен 1.

Непосредственное усмотрение истинности либо ложности условия

Если из контекста непосредственно усматривается истинность либо ложность условия A выражения "вариант$(A\ t_1\ t_2)$", то оно заменяется на соответствующее выражение t_i. Здесь рассматриваются следующие случаи:

1. Дизъюнктивный член утверждения A содержится в контексте условного выражения. Тогда оно заменяется на t_1.
2. Дизъюнктивный член отрицания утверждения A содержится в контексте условного выражения. Тогда оно заменяется на t_2.

Прием срабатывает на нулевом уровне.

Усмотрение части конъюнктивных членов условия

Если часть конъюнктивных членов утверждения A встречается в контексте условного выражения "вариант$(A\ t_1\ t_2)$", то эти члены в данном выражении отбрасываются. Уровень срабатывания равен 1.

Попытка установления истинности либо ложности условия путем решения задачи на доказательство

Пусть выражение "вариант$(A\ t_1\ t_2)$" расположено в условии задачи на преобразование, либо в задаче на исследование, целевая установка которой указывает на достаточно высокую вероятность исключения условных выражений. Тогда предпринимается попытка установить истинность либо ложность утверждения A с помощью обращений к вспомогательным задачам на доказательство. Они решаются с максимальным уровнем 4 и достаточно ограниченной трудоемкостью. При успехе проверки выражение заменяется на t_1 либо t_2. Уровень срабатывания приема равен 4 в случае задачи на преобразование и 0 в случае задачи на исследование. Прием блокируется, если условное выражение содержится под описателем "отображение", а утверждение A зависит от переменных связывающей приставки. Этот случай означает определение функции разбором случаев по значениям аргумента, и устранение "варианта" здесь маловероятно.

1.13 Символ "альтернатива"

Символ "альтернатива" практически никогда в задачах не встречается. Для него имеет единственный прием, реализованный на ЛОСе. Он заменяет утверждение "альтернатива(A $P(t_1\ \ldots\ t_{i-1}p\ t_{i+1}\ \ldots\ t_n)$ $P(t_1\ \ldots\ t_{i-1}q\ t_{i+1}\ \ldots\ t_n)$)" на утверждение $P(t_1\ \ldots\ t_{i-1}$вариант$(A\ p\ q)\ t_{i+1}\ \ldots\ t_n)$. Уровень срабатывания приема равен 1.

1.14 Приемы символов, используемых при задании конечных упорядоченных наборов

Для символов "набор", "префикс", "суффикс", "конкатенация", "поднабор", "наборномеров", позволяющих создавать конечные упорядоченные наборы элементов, на ЛОСе реализован ряд простых приемов.

Равенство двух наборов

Равенство двух наборов "набор($t_1 \ldots\ t_n$)" и "набор($p_1 \ldots p_m$)" преобразуется при $m = n$ в конъюнкцию равенств $t_1 = p_1$, …, $t_n = p_n$. При $m \neq n$ оно заменяется на логическюу константу "ложь". Прием срабатывает на уровне 0, однако блокируется, если рассматривается корневое отрицание равенства двух наборов, помеченное комментарием "набор". Такие отрицания равенств представляют собой удобную форму записи условий, которые невыгодно приводить к виду дизъюнкций. Например, условие невырожденности вектора с координатами p, q обычно записывается как $(p, q) \neq (0, 0)$.

Определение длины набора

Выражение "длинанабора(набор($t_1 \ldots t_n$))" преобразуется в десятичную запись числа n. Уровень срабатывания равен 0.

Определение элемента набора по его номеру

Выражение "значение(набор($t_1 \ldots t_n$)i)", где i - десятичная запись натурального числа от 1 до n, заменяется на t_i. Уровень срабатывания равен 0.

Устранение равенства некоторого выражения набору переменных, связанных внешними кванторами

Пусть равенство $t =$ набор$(x_1 \ldots x_n)$ имеет в своей правой части попарно различные переменные, связанные внешними кванторами и не входящие в выражение t.

Чтобы появилась возможность впоследствии применять приемы, исключающие данные связанные переменные, это равенство преобразуется к виду "и(слово(t) длина-набора(t)= n x_1 = значение(t 1) $\dots$ x_n = значение(t n))", где эти переменные явно выражены через t. Прием блокируется для некоторых ситуаций, встречающихся в аналитической геометрии (например, если t - выражение вида "коорд(...)", определяющее набор координат точки либо вектора). Уровень срабатывания равен 1.

Приемы символа "префикс"

Несколько простых приемов связаны с исключением символа "префикс". Они срабатывают на уровне 1.

1. Выражение "префикс(a набор($b_1 \dots b_n$))" преобразуется к виду "набор(a $b_1 \dots b_n$)".
2. Выражение "префикс(a пустоеслово)" заменяется на "набор(a)".
3. Выражение "префикс(a конкатенация(набор($b_1 \dots b_n$)$c_1 \dots c_m$))" заменяется на "конкатенация(набор(a $b_1 \dots b_n$)$c_1 \dots c_m$)".
4. Выражение "префикс(a конкатенация($c_1 \dots c_m$))", где выражение c_1 не имеет своим заголовком символа "набор", преобразуется к виду "конкатенация(набор(a)$c_1 \dots c_m$)".
5. Выражение "префикс(a префикс(b c))" заменяется на выражение "конкатенация(набор(a b)c)".

Приемы символа "конкатенация"

Следующие приемы символа "конкатенация" срабатывают на нулевом уровне:

1. Если в выражении "конкатенация(...)" идущие подряд операнды имеют заголовок "набор", то они объединяются в один операнд с заголовком "набор".
2. Если выражение "конкатенация(...)" расположено под такой операцией, для которой порядок элементов в наборе несущественен, то предпринимается извлечение всех операндов с заголовком "набор", объединение их в общий операнд с заголовком "набор", и перенесение данного операнда на первое место. Для срабатывания приема необходимо, чтобы либо операндов "набор(...)" было больше одного, либо чтобы единственный такой операнд не был расположен на первом месте.
3. Выражение "конкатенация(набор(a)b)", где операнд b не имеет заголовка "набор", преобразуется к виду "префикс(a b)".
4. Если в выражении "конкатенация(...)" вслед за операндом вида "набор($a_1 \dots a_n$)" расположен операнд вида "префикс(b c)", то первый операнд заменяется на "набор($a_1 \dots a_n b$)", а второй - на c.
5. Если в выражении "конкатенация(...)" операнд "префикс(a b)" не устраняется одним из описанных выше приемов, то он заменяется на два идущих подряд операнда "набор(a)", b.
6. В выражении "конкатенация(...)" отбрасываются операнды "пустоеслово".

Если выражение "конкатенация(...)" находится под операцией, для которой несущественен порядок элементов набора, то на уровне 3 предпринимается лексикографическое упорядочение ее операндов. Оно не затрагивает первого операнда "набор(...)", если таковой имеется.

Приемы символа "суффикс"

Имеется всего два таких приема, реализованных на ЛОСе. Уровень срабатывания их равен 1. Первый заменяет выражение "суффикс(набор($a_1 \dots a_n$)b)" на "набор($a_1 \dots a_n$ b)". Второй - заменяет выражение "суффикс(пустоеслово a)" на "набор(a)".

Приемы символов "поднабор", "наборномеров"

Эти символы имеют каждый по одному приему. Первый прием заменяет выражение "поднабор(набор($a_1 \dots a_n$)i j)" на выражение "набор($a_i a_{i+1} \dots a_j$)". Второй - заменяет выражение "наборномеров (i j)" на выражение "набор(i $i+1$ $\dots$ j)". Уровни срабатывания этих приемов равны 0.

1.15 Приемы символа "класс"

Общих приемов символа "класс", реализованных на ЛОСе, немного, и все они несложны. Общие приемы этого символа, реализованные на ГЕНОЛОГе, составляют значительно больший список; они собраны в разделе базы приемов, относящемся к алгебре множеств. Чтобы избежать известных парадоксов теории множеств, на использование описателя "класс" накладываются определенные ограничения. Во избежание появления слишком "больших" множеств, встречающихся в парадоксах, разрешается применять данный описатель лишь таким образом, чтобы он выделял подмножество в каком-либо ранее уже построенном множестве. Этот подход хорошо известен в аксиоматической теории множеств.

Нормализация связывающей приставки

На нулевом уровне предпринимается такое переобозначение переменных связывающей приставки описателя "класс(...)", чтобы они не имели свободных вхождений в утверждения, относительно которых рассматривается данный описатель.

Исключение описателя

Выражение "класс(x принадлежит(x t))" заменяется на t. Здесь t - выражение, не содержащее x. Уровень срабатывания приема равен 0.

Условие принадлежности классу

Утверждения "принадлежит(t класс(x $P(x)$))", "принадлежит(набор($t_1 \dots t_n$) класс($x_1 \dots x_n$ $P(x_1 \dots x_n)$))" заменяются, соответственно, на $P(t)$ и на $P(t_1 \dots t_n)$. Если в последнем случае длины набора и связывающей приставки различаются, утверждение заменяется на константу "ложь". Уровень срабатывания приема равен 0.

Отождествление двух описателей "класс", отличающихся переобозначением связанных переменных

Если два выражения "класс(...)" получаются друг из друга лишь переобозначением связанных переменных, причем эти переменные в обоих случаях уже были переобозначены так, чтобы не совпадать со своодными переменными контекстов обоих выражений, то одно из них заменяется на другое. Предпочтение отдается тому выражению, которое идет раньше в лексикографическом порядке. Уровень срабатывания приема равен 0.

Замена на константу "ложь" условия принадлежности области определения отображения, если из контекста усматривается пустота этой области

Если в задаче усматривается утверждение "равно(класс($x_1 \dots x_n$ $P(x_1 \dots x_n)$)пусто)", расположенное в контексте выражения "отображение($x_1 \dots x_n$ $P(x_1 \dots x_n)$ $t(x_1 \dots x_n)$)", то последнее заменяется на "отображение($x_1 \dots x_n$ ложь $t(x_1 \dots x_n)$)". Уровень срабатывания 1.

Расшифровка равенства описателя "класс" пустому множеству

Утверждение "равно(класс($x_1 \dots x_n$ $P(x_1 \dots x_n)$)пусто)" заменяется на "не(существует($x_1 \dots x_n$ $P(x_1 \dots x_n)$))". Попытка применения данного приема выполняется после попытки применения предыдущего. Уровень срабатывания равен 1.

Ввод вспомогательного обозначения для описателя "класс"

Если в условии задачи на описание, не имеющей целей "редакция", "прямойответ", встречается выражение "класс(x $P(x)$)", не содержащее неизвестных, то для него вводится вспомогательное обозначение. Выбирается новая переменная y, и в список посылок заносится равенство "класс(x $P(x)$))= y". Оно снабжается комментарием "ориентацияравенства", так что далее будут применяться приемы, заменяющие все вхождения в задачу данного выражения "класс(...)" на y. Перед регистрацией указанного равенства предпринимается рассмотрение утверждения $\forall_x(x \in y \to P(x))$. Оно последовательно обрабатывается двумя вспомогательными задачами на описание - сначала для развертки "по определениям", затем - для обратной свертки. Те конъюнктивные члены результата, которые представляют собой элементарные утверждения либо простые импликации, регистрируются в посылках. Они могут позволить работать далее с обозначением y напрямую, без вводящего это обозначение равенства. Уровень срабатывания приема равен 5. Заметим, что необходимое для возможности срабатывания приема отсутствие цели "прямойответ" означает, скорее всего, что задача была создана для решения внешней задачи на доказательство. В этой ситуации дополнительные посылки, сопровождающие y, могут позволить вывести необходимые для доказательства следствия.

1.16 Приемы символа "отображение"

Количество общих приемов символа "отображение" невелико - как реализованных на ЛОСе, так и на ГЕНОЛОГе.

Нормализация связывающей приставки

Прием переобозначает переменные связывающей приставки так, чтобы они отличались от свободных переменных утверждений контекста. Уровень срабатывания равен 0.

Определение значения в заданной точке

Выражения "значение(отображение(x $P(x)$ $t(x)$) A)" и "значение(отображение($x_1 \ldots x_n$ $P(x_1 \ldots x_n)$ $t(x_1 \ldots x_n)$) набор($A_1 \ldots A_n$))" заменяются, соответственно, на $t(A)$ и $t(A_1 \ldots A_n)$. Уровень срабатывания равен 0.

Область определения отображения

Выражение "область(отображение($x_1 \ldots x_n$ $P(x_1 \ldots x_n)$ $t(x_1 \ldots x_n)$))" заменяется на результат упрощения выражения "класс($x_1 \ldots x_n$ $P(x_1 \ldots x_n)$)". Упрощение предпринимается с помощью вспомогательной задачи на преобразование, решаемой до максимального уровня 4. Уровень срабатывания равен 0.

Переобозначение связанных переменных для отождествления описателей "отображение"

Если в задаче усматриваются два различных выражения "отображение(...)", отличающихся друг от друга переобозначением связанных переменных, то создается эталонная версия C указанных выражений A, B, полученная из них переобозначением переменных связывающей приставки на новые переменные. Затем все вхождения в задачу выражений A, B заменяются на C. Уровень срабатывания равен 1.

Попытка выразить через числовые параметры числовй атом, используемый для определения значения функции

Пусть в посылке задачи на доказательство либо задачи на исследование, имеющей цель "известно", встречается выражение f(отображение($x_1 \ldots x_n$ $P(x_1 \ldots x_n)$ $t(x_1 \ldots x_n)$)), имеющее внутри $t(x_1 \ldots x_n)$ вхождение числового атома A, отличного от переменной и зависящего от $x_1, \ldots, x_n$. Такой атом представляет собой числовую характеристику нечислового объекта (масса, длина, мощность и т.п.). Пусть, далее, f - числовой функционал (знак конечного суммирования, интегрирования и т.п.). Тогда предпринимается попытка выразить A через числовые параметры задачи, быть может, используя также другие числовые атомы, специальным образом связанные с A. Для этой цели вводится вспомогательная задача на описание Z', условиями которой служат равенство $A = x$ и утверждение "число(x)"; x - новая переменная, играющая роль неизвестной. К посылкам относятся все утверждения контекста рассматриваемого вхождения A. Задача имеет цель (известно ...), перечисляющую все числовые параметры указанного контекста (в том числе, возможно, некоторые переменные x_i связывающей приставки).

Если внутри числового атома A усматривается вхождение выражения "значение(g s)", где g - переменная, а s содержит все переменные $x_1, \ldots, x_n$, то предпринимается рассмотрение посылок текущей задачи Z, имеющих вхождения выражений вида "отображение($x_1 \ldots x_n$ $P(x_1 \ldots x_n)$ $q(x_1 \ldots x_n)$)". Если внутри $q(x_1 \ldots x_n)$ встречается отличный от A числовой атом B, причем B имеет подвыражение "значение(g s')", где s' содержит все переменные $x_1, \ldots, x_n$, то на время решения задачи Z' числовой атом B объявляется известным. Для этого вводится вспомогательная переменная y; список посылок задачи Z' пополняется равенством $B = y$, и y регистрируется в цели (известно ...).

Задача Z' решается до максимального уровня 5. Если на нее получен ответ, дающий выражение R числового атома A через числовые параметры, то в R восстанавливаются числовые атомы B, временно обозначенные вспомогательными переменными y (если такие переменные вообще имеются в R), и предпринимается замена в текущей посылке атома A на R. Уровень срабатывания приема равен 5.

Попытка определить значение функции в задаче на вычисление

Задачи на описание, имеющие цель "вычисление", предназначены для составления ЛОС-программ, вычисляющих значения неизвестных задачи по значениям ее известных параметров. Такая программа создается в два этапа: сначала условия задачи на описание преобразуются к виду, понятному компилятору ГЕНОЛОГа, а затем предпринимается обращение к компилятору. На первом этапе иногда оказывается достаточно использовать стандартные схемы вычислений для чисто технической переформулировки условий (представление интеграла в виде конечной суммы, переход от дифференциального уравнения к конечно-разностной схеме и т.п.). Иногда же приходится сначала решать обычную задачу на явное выражение неизвестных через известные, и лишь затем переходить к техническим преобразованиям. Такая ситуация складывается, например, при составлении программ, относящихся к физическим процессам, заданным на логическом языке. Типичным явлением здесь является определение траектории какого - либо параметра, т.е. таблицы значений, принимаемых им в последовательные моменты времени. В этом случае исходная задача на описание имеет условие вида "$y =$ отображение(t $P(t)$ $F(t)$)"; y - неизвестная для траектории.

Ее решение сводится к выводу следствий в блоке анализа, где описываемый прием и обнаруживает содержащую выражение "отображение(t $P(t)$ $F(t)$)" посылку. Если выражение $F(t)$ содержит неизвестные блока анализа (например, задано через косвенные характеристики процессов - координаты, скорости, силы, пока еще не выраженные явно через числовые параметры), то создается вспомогательная задача на описание Z'. Ее условиями становятся утверждения $x = F(t)$, "число(x)", посылками - все утверждения из контекста $F(t)$. Неизвестной задачи служит новая переменная x; цель (известно ...) содержит все известные параметры блока анализа и переменную t. Если удается получить ответ на задачу Z', дающий явное выражение R для ее неизвестной, то предпринимается замена в рассматриваемой посылке выражения $F(t)$ на R. Уровень срабатывания приема равен 2.

1.17 Приемы логических констант

Приемы логических констант обычно выполняют их исключение либо усматривают завершение процесса решения задачи.

Условие "истина" задачи на доказательство

Если условием задачи на доказательство служит логическая константа "истина", то выдается ответ "истина". Уровень срабатывания равен 0.

Условие "истина" задачи на описание

Если логическая константа "истина" служит условием задачи на описание, то рассматриваются два случая. При отсутствии прочих условий выдается ответ "истина" (не логический символ, как в случае задачи на доказательство, а однобуквенный терм). Иначе условие "истина" удаляется из списка условий, и решение продолжается. Если при этом остается единственное условие с неизвестными, вес его понижается до 0. Данная мера нужна, чтобы срабатывали приемы усмотрения ответа, которые в противном случае "не заметят" исчезновения (преобразования в константу "истина") неизвестных условий, ранее не вписывавшихся в требуемый вид ответа. Уровень срабатывания приема равен 0.

Посылка "истина"

Если логическая константа "истина" служит посылкой задачи, причем имеются и другие посылки, то она удаляется. Уровень срабатывания 0.

Консеквент "истина"

Если логическая константа "истина" является консеквентом кванторной импликации, то эта импликация заменяется на константу "истина". Уровень срабатывания 0.

Антецедент "истина"

Если логическая константа "истина" является антецедентом кванторной импликации, то она отбрасывается. При этом могут быть отброшены также частицы "если", "то". Уровень срабатывания 0.

Логическая константа "истина" под дизъюнкцией либо квантором существования

Если константа "истина" является операндом дизъюнкции либо квантора существования, то последние заменяются на эту константу. Уровень срабатывания 0.

Логическая константа "истина" под конъюнкцией либо эквивалентностью

Если константа "истина" является операндом конъюнкции либо эквивалентсности, то этот операнд отбрасывается. Уровень срабатывания 0.

Логическая константа "истина" под отрицанием

Отрицание константы "истина" заменяется на "ложь". Уровень срабатывания 0.

Логическая константа "истина" в условном выражении

Выражение "вариант(истина t_1 t_2)" заменяется на t_1. Уровень срабатывания 0.

Условие "ложь" задачи на доказательство

Если условием задачи на доказательство оказалась константа "ложь", то на уровне 0 выдается отказ. Однако, может так оказаться, что список посылок задачи на доказательство, имеющей условие "ложь", противоречив. В этом случае допустима выдача ответа "истина". Чтобы не спешить с выдачей отказа до того, как предприняты попытки установить противоречивость посылок, предсумотрен комментарий "ложь". При его наличии выдача отказа откладывается до уровня 5.

Условие "ложь" задачи на описание

Если константа "ложь" оказалась условием задачи на описание, то при наличии целей "полный" и "пример" выдается отказ. Иначе выдается ответ "ложь". Уровень срабатывания равен 0.

Посылка "ложь" задачи на доказательство

Если задача на доказательство имеет посылку "ложь", то на нее выдается ответ "истина". Уровень срабатывания 0.

Посылка "ложь" задачи на описание

Если задача на описание имеет посылку "ложь", то на нее выдается ответ "истина". Уровень срабатывания 0.

Посылка "ложь" задачи на преобразование

Если задача на преобразование имеет посылку "ложь", то на нее выдается ответ - однобуквенный терм "противоречие". Символ "противоречие" может возникать в решателе только указанным образом, и в особых случаях нужно учитывать возможность его появления как ответа задачи на преобразование. Уровень срабатывания равен 0.

Посылка "ложь" задачи на исследование

Если задача на исследование является блоком анализа внешней задачи на описание, то при появлении в ее посылках константы "ложь" предпринимаются регистрация этой константы в условиях задачи на описание и возвращение к сканированию последней. Если задача на исследование решалась с целью усмотрения противоречивости списка посылок, то при обнаружении в ней константы "ложь" просто происходит возвращение к внешней задаче. Уровень срабатывания равен 0.

Логическая константа "ложь" под отрицанием либо в антецеденте кванторной импликации

Если утверждение представляет собой отрицание константы "ложь" либо кванторную импликацию с антецедентом "ложь", то оно заменяется на константу "истина". Уровень срабатывания 0.

Логическая константа "ложь" под конъюнкцией либо под квантором существования

Если утверждение представляет собой конъюнкцию с операндом "ложь" либо квантор существования с подкванторным утверждением "ложь", то оно заменяется на константу "ложь". Уровень срабатывания 0.

Логическая константа "ложь" под дизъюнкцией

Если константа "ложь" оказывается операндом дизъюнкции, то она отбрасывается. Уровень срабатывания 0.

Логическая константа "ложь" в консеквенте кванторной импликации

Если логическая константа "ложь" является консеквентом кванторной импликации $\forall_{x_1 \ldots x_n}(A_1 \ \& \ldots \& \ A_n \to A_0)$, то она заменяется:

1. При $n = 0$ - на константу "ложь";
2. При $n > 0$ - на $\forall_{x_1 \ldots x_n}(A_1 \ \& \ldots \& \ A_{n-1} \to \neg A_n)$.

Прием срабатывает на уровне 0.

Логическая константа "ложь" под эквивалентностью

Утверждение "эквивалентно(A ложь)" заменяется на $\neg A$. Уровень срабатывания 0.

1.18 Приемы основных символов, связанных с функциями

Для символов "область", "значение", "подфункция", "существперем" на ЛОСе реализована сравнительно большая группа приемов. Напомним, что посредством "значение(f t)" обозначается значение функции f в точке t и что это выражение прорисовывается формульным редактором в виде $f(t)$.

Функция, для которой рассматривается лишь ее область определения

Если связанная переменная x квантора общности либо существования встречается только в выражениях "область(x)", за исключением, быть может, утверждения "функция(x)", расположенного в контексте указанных выражений, то утверждение "функция(x)" заменяется на "множество(x)", а все выражения "область(x)" заменяются на x. Уровень срабатывания приема равен 1.

Условие принадлежности области определения функции, заданной описателем "отображение"

Если в контексте утверждения "принадлежит(t область(f))" встречается равенство "$f =$ отображение(x $P(x)$ $s(x)$)", то данное утверждение заменяется на $P(t)$. Прием применяется и в векторной ситуации - если связывающая приставка x имеет более одной переменной, а выражение t имеет вид "набор(...)". Уровень срабатывания 1.

Замена квантора по функции, определенной на конечном множестве, на квантор по группе переменных - значений функции на элементах этого множества

Пусть связанная переменная f квантора общности либо существования встречается в одном из следующих контекстов:

1. В выражении вида $f(i)$, где i - десятичное число;
2. В утверждении "слово(f)" - антецеденте кванторной импликации либо конъюнтивном члене под квантором существования;
3. В утверждении "равно(длинанабора(f)m)", где m - десятичное число.

Тогда находится список S всевозможных выражений $f(i)$, содержащих f, и выбирается список P новых переменных, которые будут использоваться вместо данных выражений. Затем предпринимается замена под квантором выражений списка S на соответствующие им переменные списка P, замена в кванторной приставке переменной f на переменные P, и отбрасывание утверждений "слово(f)", "равно(длинанабора(f)m)". Уровень срабатывания равен 1.

Определение значения многоместной функции, заданной описателем "отображение"

Выражение "значение(f набор($t_1 \dots t_n$))", в контексте которого встречается равенство "f = отображение($x_1 \dots x_n$ $P(x_1 \dots x_n)$ $g(x_1 \dots x_n)$)", заменяется на результат упрощения выражения $g(t_1 \dots t_n)$. Прием применяется только к условиям задач и в ряде специальных случаев блокируется (например, если указанное выражение является операндом описателя "отображение"). Уровень срабатывания равен 1. Заметим, что в случае одноместных функций аналогичные преобразования выполняются приемами, заданными на ГЕНОЛОГе.

Исключение символа "подфункция"

Выражение "подфункция(f набор($i_1 \dots i_k$) набор($t_1 \dots t_k$))" обозначает результат фиксации аргументов многоместной функции f, номера которых суть $i_1, \dots, i_k$, значениями $t_1, \dots, t_k$. Если f имеет вид "отображение($x_1 \dots x_n$ P g)", либо в контексте содержится равенство f такому выражению, то определяются результаты A, B подстановки выражений $t_1, \dots, t_k$ вместо $x_{i_1}, \dots, x_{i_k}$ в утверждение P и выражение g. Утверждение A упрощается с помощью вспомогательной задачи на преобразование, решаемой до уровня 4; выражение B обрабатывается нормализаторами общей стандартизации. После этого находится результат $y_1, \dots, y_{n-k}$ исключения из связывающей приставки $x_1, \dots, x_n$ переменных с номерами $i_1, \dots, i_k$, и выражение "подфункция(...)" заменяется на "отображение($y_1 \dots y_{n-k}$ A B)". Прием применяется на уровне 0.

Кванторная расшифровка условия существенности переменной

Утверждение "существперем(i f)" означает, что переменная x_i функции $f(x_1 \dots x_n)$ является существенной, т.е. что найдутся такие значения остальных переменных, при подстановке которых функция не обращается в константу. Если данное утверждение является условием задачи на описание, причем f имеет вид "отображение($x_1 \dots x_n$

$P\ F)$", то оно заменяется на утверждение существования двух различных значений аргумента x_i (с одинаковыми значениями прочих переменных), при которых значения функции различаются. Уровень срабатывания приема равен 4. Особо рассматривается случай функций алгебры логики, где кванторное утверждение приобретает несколько более простой вид.

Тождественная замена с помощью кванторного тождества, позволяющая исключить символ "значение"

Для переменных f, имеющих своими значениями функции, в посылках задачи часто указываются кванторные тождества $\forall_{x_1 \ldots x_n}(A_1\ \&\ \ldots\ \&\ A_n \rightarrow B = C$, позволяющие определить значение C выражения B, содержащего подтерм вида "значение($f\ r$)". Имеется прием, предпринимающий попытку воспользоваться таким тождеством для исключения подтермов указанного вида. Он активизируется в задаче на исследование при обнаружении вхождения подтерма "значение($f\ r$)" в консеквент кванторного тождества. Проверяется, что кванторная приставка пересекается с параметрами выражения r и что либо терм C не содержит символа "значение", либо он не содержит неизвестных, в то время как терм B содержит неизвестные. Проверяется, что выражение B не содержит связанных переменных. Затем просматриваются такие посылки текущей задачи на исследование, в которых можно выделить подтерм "значение(f R)", получаемый подстановкой в "значение($f\ r$)" некоторых термов $t_1, \ldots, t_n$ вместо переменных $x_1, \ldots, x_n$. Проверяется, что эта подстановка переводит антецеденты кванторной импликации в очевидные утверждения, после чего подтерм "значение(f R)" заменяется в посылке на результат применения подстановки к C. Уровень срабатывания приема равен 4. Блокируется преобразование посылок, специально введенных для указания связей значения функции f с прочими объектами и помеченных комментарием "узелвывода".

Применение кванторного тождества, помеченного комментарием "блокзамен"

Чтобы кванторное тождество вида $\forall_{x_1 \ldots x_n}(A_1\ \&\ \ldots\ \&\ A_n \rightarrow f(t) = r$, расположенное в посылках текущей задачи, использовалось для повсеместного исключения выражений вида $f(\ldots)$, оно может быть помечено комментарием (блокзамен f). Исключение выполняется двумя приемами, аналогичными приему предыдущего пункта. Первый из них инициализирует использование тождества. Как только тождество возникает, прием предпринимает просмотр посылок и применяет тождество к их подвыражениям $f(\ldots)$. Второй прием необходим, чтобы тождество могло быть применено к тем выражениям $f(\ldots)$, которые появляются в задаче после применения первого приема. Он активизируется при усмотрении выражения вида "значение($f\ R$)"; предпринимает поиск тождества указанного выше вида, помеченного комментарием (блокзамен f), и применяет его к данному выражению. Уровень срабатывания обоих приемов равен 2.

Попытка получить рекуррентную формулу для числового атома, содержащего подвыражение с заголовком "значение"

Если посылка задачи на исследование имеет числовой атом C, содержащий подвыражение "значение($f\ t$)", где t неконстантное, то предпринимается попытка получить для C рекуррентную формулу. Для этого находится кванторная посылка

$\forall_x(A_1 \ \& \ \ldots \ \& \ A_n \to A_0)$, консеквент которой имеет подвыражение "значение(f r)". Проверяется, что антецеденты этой посылки определяют либо конечный отрезок целочисленных значений переменной x, либо множество всех ее натуральных значений, не меньших заданного. В консеквенте A_0 находится еще одно подвыражение "значение(f R)" и проверяется, что r - числовое выражение, большее на единицу выражения R. Если такая кванторная посылка обнаружена, то определяются результаты C_1, C_2 замены в терме C подвыражения "значение(f t)" на выражения "значение(f R)", "значение(f r)" соответственно. Выбираются новые переменные y, z, и создается вспомогательная задача на описание Z'. Ее посылками служат все посылки текущей задачи, пополненные антецедентами $A_1, \ldots A_n$ и утверждениями "число(y)", $y = C_1$. Условия суть утверждения "число(z)", $z = C_2$. Задача имеет цели "полный", "явное", "прямойответ", "рекурсия", "упростить". Ее единственной неизвестной служит z; цель (известно ...) перечисляет все известные параметры текущей задачи и переменную y. Если на задачу Z' получен ответ P, то находится результат P' подстановки в него вместо переменных y, z выражений C_1, C_2. После этого в список посылок заносится утверждение $\forall_x(A_1 \ \& \ \ldots \ \& \ A_n \to P')$, дающее рекуррентную зависимость для числового атома C. Уровень срабатывания приема равен 3. От задачи на исследование дополнительно требуется наличие цели "известно".

Понижение весов кванторных импликаций, используемых для вывода утверждений с символом "значение"

Если задача на преобразование имеет цель (значение x), то при ее решении стимулируется вывод следствий в посылках, содержащих выражения вида "значение(f x)". Имеется прием, активизируемый при обнаружении в условии такой задачи выражения вида "значение(A x)" и понижающий до уровня 2 веса всех кванторных импликаций в посылках, содержащих подтерм "значение(A ...)". Этот прием срабатывает на уровне 3.

Вывод следствий, содержащих заданный терм "значение(f t)"

Начиная с данного пункта, будет рассмотрена серия приемов, относящихся к задачам на описание, имеющим цель "длялюбого". Как уже говорилось выше, такая задача Z' вводится для анализа кванторного условия F внешней задачи на описание Z. В посылки ее заносятся все антецеденты импликации F, а конъюнктивные члены консеквента становятся условием. Задача не имеет неизвестных, но имеет цель (независит ...), перечисляющую переменные кванторной приставки F. При ее решении можно пополнять список посылок дополнительными утверждениями существования, не являющимися следствиями исходных. Все такие дополнения регистрируются в комментарии (блокпосылок A_1 A_2). Здесь A_1 - набор троек (X_1 X_2 P), соответствующих новым посылкам "существует(X B)". Связывающая приставка X разбита на две компоненты X_1, X_2, причем переменные первой из них считаются не зависящими от переменных кванторной приставки импликации F, а переменные второй - могут зависеть. Данное разбиение оставляется на усмотрение того приема, который добавляет посылку. P - набор конъюнктивных членов утверждения B. A_2 - набор конъюнктивных членов результата упрощения утверждения, выражающего, что для любых удовлетворяющих антецедентам импликации F значений переменных ее кванторной приставки существуют значения переменных из списков X_2, при которых истинна конъюнкция всех утверждений P. Различные комментарии (блокпосылок ...) относятся к несвязанным между собой группам переменных X_i.

Если условие задачи на описание, имеющей цель "длялюбого", содержит подтерм "значение(f t)", у которого выражение t зависит от переменных цели (независит ...), то необходимо попытаться исключить данный подтерм из условия. Для этого нужно вывести какие-то следствия, содержащие подтерм. В посылках задачи ищется кванторная импликация K, у которой консеквент имеет подвыражение "значение(f r)". Определяется список S_1 всех переменных кванторной приставки импликации K, входящих в r, и проверяется, что t является результатом подстановки в r некоторых термов T_1 вместо переменных S_1. Находится оставшаяся часть S_2 кванторной приставки импликации K. Для ее переобозначения выбирается список S_3 не используемых в задаче переменных. Затем находятся: список D результатов подстановки T_1, S_3 вместо S_1, S_2 в антецеденты импликации K, а также результат E применения данной подстановки к консеквенту K. Проверяется, что E не имеет подтермов "значение(...)", у которых второй операнд содержит переменные списка S_3. Таким образом, оказывается выполнена подготовка к регистрации в посылках нового утверждения E, содержащего подтерм "значение(f t)". Однако, для его вывода требуется истинность всех утверждений списка D, на которые навешен квантор существования по переменным S_3. Эти утверждения регистрируются в указанном выше комментарии (блокпосылок ...), и лишь затем к посылкам задачи присоединяются утверждение E и все утверждения D. Прием срабатывает на уровне 2.

Вывод промежуточных импликаций в задаче с целью "длялюбого"

В предыдущем приеме перед выводом следствия E проверялось, что в нем отсутствуют термы "значение(g s)", у которых s содержит переменные списка S_3. Если такие термы в E имеются, то прием несколько корректируется - вместо E выводится кванторная импликация, у которой E является консеквентом. Кванторную приставку импликации образуют все переменные списка S_3, входящие в указанные выражения s. Антецедентами служат все утверждения D, зависящие от таких переменных. Перед выводом проверяется, что прочие утверждения списка D являются следствиями списка посылок текущей задачи. Выведенная импликация снабжается комментарием "чисткапосылок". Так как в данном случае выводимое утверждение тоже является следствием посылок, то никакой регистрации его в комментариях (блокпосылок ...) не выполняется. Кванторная импликация, выведенная приемом, называется промежуточной. Бескванторные следствия из нее выводит прием, описываемый в следующем пункте. Заметим, что применение к промежуточным импликациям приема из предыдущего пункта блокируется. Примеры применения приема можно найти в простейших задачах на доказательство из раздела задачника "Математический анализ". Уровень срабатывания его, как и предыдущего приема, равен 2.

Извлечение следствий из промежуточных кванторных импликаций в задаче с целью "длялюбого"

Промежуточная кванторная импликация K, введенная предыдущим приемом, содержит своем консеквенте подтерм "значение(g s)". Чтобы как-то связать это значение с другой информацией о функции g, используется еще одна кванторная импликация K' из посылок текущей задачи, содержащая g. Предварительно проверяется, что все переменные кванторной приставки K содержатся в s и что утверждение K не имеет других подтермов вида "значение(...)", второй аргумент которых пересекается с кванторной приставкой K. Проверяется также, что ни одна из импликаций K, K' не была использована при выводе другой. Перед выводом следствий предпринимается такое переобозначение связанных переменных обоих импликаций, чтобы они отличались друг от друга и от переменных прочих утверждений задачи. В консеквенте импликации K' находится подтерм "значение(g u)". Составляется список H всех переменных кванторной приставки K', встречающихся в u, и проверяется отсутствие в K' подтермов "значение(...)", содержащих не вошедшие в H переменные кванторной приставки. Далее составляется список H', полученный добавлением к H кванторной приставки импликации K. Находятся такие термы T, подстановка которых вместо H' унифицирует выражения u и s. Эта подстановка определяет значения всех переменных кванторной приставки K и части переменных кванторной прставки K'. Находятся результаты C_1, C_2 ее применения к консеквентам импликаций K, K', а также список L результатов ее применения к антецедентам обеих импликаций. Находится результат U применения данной подстановки к u. Составляется список V переменных связывающих приставок импликаций K, K', встречающихся в утверждениях из L. Он подразбивается на два подсписка V_1, V_2, первый из которых образован всеми переменными, не входящими в U. Чтобы получить следствия C_1, C_2 из K, K', нужно ввести дополнительную посылку - утверждение о существовании значений переменных списка V, при которых истинны антецеденты L. Это утверждение регистрируется в комментарии (блокпосылок ...), причем переменные списка V_1 считаются при регистрации не зависящими от переменных кванторной приставки кванторного условия F внешней задачи, а переменные V_2 - зависящими. Последние переменные присоединяются также к цели (независит ...). Завершает действия приема присоединение к посылкам утверждений C_1, C_2, а также всех утверждений списка L. Уровень срабатывания приема равен 2.

Использование кванторных импликаций для разбора случаев в задаче с целью "длялюбого"

Если в условии задачи на описание, имеющей цель "длялюбого", встречается выражение "значение(f t)", у которого t содержит переменные, выделенные целью (независит ...), то предпринимается поиск в посылках кванторной импликации K, имеющей в своем консеквенте выражение "значение(f r)". Проверяется, что эта кванторная импликация не была использована для получения следствия с выражением "значение(f t)" и не является промежуточной импликацией (см. выше). Составляется список S_1 всех переменных кванторной приставки K, входящих в r, и определяются такие термы T, подстановка которых в r вместо S_1 дает t. Находится список S_2 остальных переменных кванторной приставки K, и для их переобозначения выбираются не используемые в задаче переменные S_3. Далее определяются список D результатов подстановки T, S_3 вместо S_1, S_2 в антецеденты импликации K, а также результат E применения этой подстановки к консеквенту K. Проверяется, что E не содержит подвыражений "значение(...)", второй операнд которых пересекается с S_3. Список D разбивается на два подсписка D_1, D_2, к первому из которых относятся все утверждения, имеющие выделенные целью (независит ...) переменные. Проверяется, что D_1 непуст и что все переменные списка S_3 содержатся в D_2. Далее в комментарии (блокпосылок ...) регистрируется утверждение о существовании таких значений переменных списка S_3, при которых истинны все утверждения из D_2. Однако, этого утверждения еще не достаточно для вывода следствия E, так как остается нереализованным список D_1. Поэтому находится конъюнкция H всех утверждений списка D_1, и в список посылок текущей задачи заносится дизъюнкция $(E \ \& \ H) \vee \neg H$. Она помечается комментарием "разборслучаев". Заметим, что задача имеет цель (независит ...), а альтернативы новой посылки зависят от переменных данной цели. Чтобы эта зависимость не перешла в ответ задачи, он будет формироваться как конъюнкция ответов, найденных для подслучаев (см. выше прием разбора случаев по дизъюнктивной посылке задачи на описание). Такая схема рассуждений является типичной, например, при получении оценок. Сначала неравенства для оцениваемой величины x находятся в каждом подслучае, а затем конъюнкция этих неравенств дает оценку, верную во всех ситуациях. Прием срабатывает на уровне 3.

Попытка использовать константную функцию

Если в условии задачи на описание, имеющей цель "длялюбого", встречается выражение "значение(f t)", у которого t содержит переменные, выделенные цель (известно ...), а f есть неизвестная внешней задачи на описание, то предпринимается поптыка подобрать в качестве f константную функцию. Для этого сначала упрощается выражение "область(f)". Если результат упрощения M не содержит переменной f, то создается выражение F вида "отображение(x $x \in M$ a)". Здесь x, a - новые переменные. Находится список S свободных переменных равенства $f = F$, и в комментарии (блокпосылок ...) регистрируется утверждение о существовании таких значений переменных S, при которых это равенство истинно. Затем равенство $f = F$ заносится в список посылок, и во всех условиях задачи выражения "значение(f X)" заменяются на a. Прием срабатывает на уровне 5. Разумеется, для попытки подбора константной функции нужны дополнительные веские основания; нужно обеспечить возможность отката при неудаче и т.п. Однако, на рассматривавшихся задачах достаточно было и предлагаемой версии. Данный и другие приемы, относящиеся к задачам с целью "длялюбого", проверены пока на очень небольшом обучающем материале. Они имеют характер грубых предварительных версий и требуют существенной последующей доработки. Тем не менее, они все же иллюстрируют общие принципы развития аппарата, связанного с кванторными условиями задач на описание.

Использование кванторной импликации для получения оценки выражения "значение(f t)"

Если посылка K задачи на описание, имеющей цель "длялюбого", представляет собой кванторную импликацию с неравенством для выражения "значение(f x)" в консеквенте, то предпринимается попытка вывести с ее помощью неравенство для встречающегося в условии P выражения "значение(f t)". Проверяется, что t зависит от переменных, упоминаемых в цели (независит ...) текущей задачи и что x - переменная кванторной приставки K. Далее проверяется, что условие P само имеет вид неравенства, причем направление неравенства для "значение(f t)", получаемого из K, позволяет воспользоваться им для вывода P "по монотонности" из нового условия Q, полученного заменой выражения "значение(f t)" на предлагаемую неравенством оценку этого выражения. Рассматривается кванторная приставка X импликации K. Если она состоит из единственной переменной x, то для подстановки вместо X берется одноэлементный список, состоящий из терма t. В противном случае проверяется, что X двухэлементна, и для подстановки вместо x снова выбирается t, а для подстановки вместо другой переменной - какое-либо выражение r, не имеющее общих переменных с целью (независит …) и такое, что "значение(f r)" встречается в условиях задачи. Находятся результаты применения указанной подстановки к антецедентам импликации K и проверяется, что они вытекают из посылок задачи. Затем определяется результат применения подстановки к консеквенту импликации, который и регистрируется в посылках задачи. Вес условия с выражением "значение(f t)" понижается до 3. Уровень срабатывания приема равен 4.

Подбор примера с заменой функциональной неизвестной на обычную

Пусть в задаче на описание с целью "пример" имеется такая неизвестная f, которая встречается в условиях "функция(f)", "равно(область(f)A)", а кроме этого - только внутри выражений "значение(f x)", расположенных в консеквентах кванторных импликаций, имеющих антецедент "принадлежит(x A)"; x - переменная кванторной приставки импликации, не встречающаяся в прочих антецедентах. Такие импликации задают некоторое общее условие на произвольное значение функции f, так что всегда можно в качестве f брать константную функцию, определенную на A и имеющую такое значение a, которое удовлетворяет данному условию. Прием, усматривающий описанную ситуацию, составляет список S утверждений, полученных из содержащих f кванторных импликаций удалением антецедента "принадлежит(x A)", исключением из кванторной приставки переменной x и заменой "значение(f x)" на "значение(f y)". Если не только f, но и некоторые другие функциональные неизвестные g удовлетворяют перечисленным выше ограничениям, причем их вхождения "значение(g x)" расположены в тех же кванторных импликациях и имеют те же самые аргументы x, то они вместе с f образуют список X преобразуемых неизвестных. Утверждения списка S преобразуются путем замены выражений "значение(g y)" для g из X на переменные g. После этого создается вспомогательная задача Z' на описание, списком условий которой служит S. Посылки ее составлены из посылок текущей задачи, всех не содержащих переменных X условий текущей задачи, а также из утверждения $y \in A$. Неизвестные суть переменные списка X. Если после решения задачи Z' получаются значения $t_1, \ldots, t_n$ для составляющих список X неизвестных $x_1, \ldots, x_n$, то в текущей задаче отбрасываются все условия, содержащие эти неизвестные, а вместо них регистрируются утверждения "равно(x_i отображение(y принадлежит(y A) t_i))" и сопровождающие конъюнктивные члены ответа задачи Z'. Уровень срабатывания приема равен 2.

Неизвестная последовательность, входящая в условия с одним и тем же аргументом

Если в задаче на описание с целью "пример" имеется неизвестная f, входящая в условие "последовательность(f A)", а кроме этого встречающаяся только внутри вхождений одного и того же выражения "значение(f t)", не связанного внешними кванторами и описателями, то применяется прием, аналогичный предыдущему - функциональная неизвестная f преобразуется в обычную. Для этого создается новая задача на описание Z', имеющая своими условиями утверждение $f \in A$, а также утверждения, полученные из отличных от "последовательность(f A)" условий текущей задачи заменой вхождений выражения "значение(f t)" на f. Посылки и цели задачи Z' - те же, что у текущей задачи. Если на нее получен отличный от символа "отказ" результат R, определяющий значение s неизвестной f, то выбирается новая переменная y, и значение s заменяется в R на "отображение(y натуральное(y) s)".

Если текущая задача не имела комментария (контекст ...), сохраняющего значения внешних неизвестных, выраженных через оставшиеся неизвестные, то сразу выдается ответ R. Иначе условия текущей задачи заменяются на конъюнктивные члены утверждения R, и предпринимается обращение к процедуре "редакторответа", обеспечивающей надлежащий учет комментария (контекст ...). Уровень срабатывания приема равен 2.

Попытка подобрать значение неизвестной из сопоставления аргументов одной и той же функции

Если задача на описание имеет цели "полный", "пример", причем ее единственная неизвестная x встречается в подвыражении условия "значение(f x)", а посылки имеют подвыражение "значение(f t)", то предпринимается попытка проверить, не подойдет ли для x значение t. Проверки истинности результатов подстановки t вместо x в условия задачи предпринимаются с достаточно сильным ограничением на трудоемкость. Если они успешно реализованы, выдается ответ $x = t$. Уровень срабатывания приема равен 4.

1.19 Примеры и упражнения

Большинство из приведенных выше приемов выполняют преобразования, целесообразность которых очевидна. Однако, в ряде случаев она может показаться сомнительной, а то и вовсе непонятной. В этих ситуациях полезно обратиться к задачнику системы, найти те задачи, в которых прием срабатывает, и с помощью отладчика ЛОСа проанализировать его действия подробнее. Иногда это может прояснить смысл преобразований. В тех редких случаях, когда и рассмотрение примера оставляет сомнения относительно целесообразности приема, следует относиться к нему как к временной "заглушке", созданной по немногим, чаще всего - по единственному примеру. Здесь можно смело браться за доработку приема или даже найти какую-то совсем иную схему решения задачи, позаботившись в первую очередь о достаточно богатом обучающем материале для анализа аналогичных ситуаций. Этот материал должен предоставить информацию о "скрытых" особенностях задачи, которые необходимо учитывать перед принятием решения о применении приема, а также обеспечить тестирование его фактического поведения. Если обновленный или вновь созданный прием имеет сравнительно общий характер и может срабатывать в различных предметных областях, то следует проявлять особую осторожность при занесении его в базу приемов, выполняя прогонку решателя по соответствующим разделам или даже по всему задачнику.

1.19.1 Прием разбора случаев по дизъюнктивному условию задачи на описание

Чтобы проиллюстрировать технику применения отладчика для анализа действий приемов, рассмотрим несколько простых примеров. В качестве первого примера выберем часто встречающийся прием разбора случаев по дизъюнктивному условию задачи на описание. Он находится в подразделе "Приемы решателя" - "Общие приемы" - "Дизъюнкция" - "Разбор случаев по дизъюнктивному условию задачи на описание".

Если задача, в которой срабатывает прием, неизвестна, то для отслеживания моментов его применения удобнее всего вставить в программу приема контрольную точку - оператор "трассировка(стоп 0)". При выходе на этот оператор программа будет останавливаться и включать отладчик ЛОСа, который и позволит рассмотреть контекст срабатывания. Если прием применяется крайне редко, то для поиска задач, где он срабатывает, удобно применять режим "прогонки" решателя по задачнику, предварительно введя указанную контрольную точку редактором ЛОСа.

Если задача, в которой прием срабатывает, известна, или такие задачи идут в рассматриваемом разделе задачника достаточно часто, то можно не пользоваться редактором ЛОСа, а запускать решение задачи через точку оглавления программ, которая соответствует рассматриваемому приему либо некоторому этапу его работы. В нашем случае лучше воспользоваться именно таким подходом. Разбор случаев часто встречается, например, при решении тригонометрических уравнений. Войдем в раздел задачника "Элементарная алгебра" - "Решение уравнений" - "Тригонометрические уравнения" - "Уравнения с неизвестной в знаменателе - 1" и выберем первую же задачу данного раздела. Чтобы перейти в оглавление программ, нажимаем "л", затем входим в указанный выше подраздел оглавления, соответствующий нашему приему. В этом подразделе выделено несколько концевых пунктов, и можно в качестве точки прерывания выбрать любой из них. Наиболее информативным, видимо, является пункт 4 - "Рассмотрение подслучая для дизъюнктивного условия". Он позволяет рассмотреть вспомогательную задачу, соответствующую текущему подслучаю, до входа в ее решение. При необходимости можно будет войти в трассировку подслучая, либо пропустить ее и сразу посмотреть ответ. Поэтому клавишами "курсор вниз - вверх" выделяем данный пункт. Для запуска решения задачи с прерыванием на нем нажимаем "курсор вправо". Начинается решение задачи, и почти сразу возникает кадр отладчика ЛОСа. В нем выделен малиновым цветом текущий (еще не выполненный) оператор "уровеньобращения(x7)". Оператору предшествует контрольная точка "прием(3 3)", соответствующая выбранному концевому пункту оглавления. Чтобы на экране прорисовать весь текст фрагмента программы, нажимаем "ф". Тогда становится виден оператор "равно(x18 ответзадачи(x17))", обращающийся к рассмотрению текущего подслучая. Для анализа текущего контекста можно выполнить следующие действия:

1. Посмотреть на дизъюнкцию, по которой проводится разбор случаев. Ее текущее вхождение является значением программной переменной x2, так что достаточно нажать "х" (кир.), "2", "Enter". Появляется формула $\sin x + \cos x = 0 \ \vee \ \cos(2x) = -1/2$.

2. Посмотреть всю текущую задачу. Для этого нажимается клавиша "з". На экране появляется текст "Найти x", под которым расположены условие $\neg(\cos x = 0)$ и приведенная выше дизъюнкция. Чтобы вернуться в кадр отладчика ЛОСа, нажимается клавиша "ф". Не следует в данной ситуации нажимать Esc, так как это приведет к обрыву трассировки и возвращению к просмотру исходного текста задачи.

3. Посмотреть текущий рассматриваемый подслучай. Для этого, используя клавиши "Home", "End", перемещаемся вдоль цепочки фрагментов текущей программы, предшествующих текущему фрагменту. Нажатие клавиши "Home" переводит в направлении к началу программы, нажатие "End" - возвращает на

один шаг в направлении от начала программы к текущему фрагменту. В одном из фрагментов данной цепочки замечаем оператор "равно(x14 набороперандов(x2))", который создает список x14 дизъюнктивных членов рассматриваемой дизъюнкции. Двигаясь от него по направлению к текущему фрагменту (клавиша "End"), обнаруживаем в следующем фрагменте оператор "позиция(x15 x14)". Очевидно, он и выделяет текущее рассматриваемое вхождение списка x14. За ним расположен оператор "равно(x16 буква(x15))", присваивающий текущий дизъюнктивный член переменной x16. Можно посмотреть значение этой переменной - $\sin x + \cos x = 0$.

4. Посмотреть задачу, созданную для обработки подслучая. Оператор "равно(x18 ответзадачи(x17))" подсказывает, что данная задача присвоена переменной x17. Так как задача представляет собой достаточно сложную структуру данных, удобно использовать сквозной ее просмотр. Поэтому нажимаем клавиши "K" (кир., заглавная); "1"; "7"; "Enter". Возникает список пронумерованных объектов - разрядов набора, представляющего задачу. Сначала идет тип задачи "описать", затем - список посылок, список весов посылок, и т.д. вплоть до 9-го элемента - блока анализа (пока отсутствующего). Выделяем клавишами "курсор вверх - вниз" для просмотра пятый элемент - список условий, после чего нажимаем "курсор вправо". Появляется список из трех утверждений $\neg(\cos x = 0)$, x – число, $\sin x + \cos x = 0$, в котором последнее утверждение - дизъюнктивный член, замещающий дизъюнкцию внешней задачи. Для возвращения в кадр отладчика ЛОСа нажимаем несколько раз "End" либо "Esc". Лучше избегать нажатий последней клавиши, так как лишнее ее нажатие оборвет трассировку.

Используя средства отладчика, можно посмотреть и другие сведения о текущем контексте - например, комментарии к вспомогательной задаче, ее цели и т.п. Продолжим трассировку работы приема, посмотрев результат решения задачи x17. Для этого нужно переустановить режим трассировки на пооператорные шаги в текущем кадре - нажать клавишу "2". Если этого не сделать, а продолжить трассировку нажатиями клавиши "Enter", то отладчик будет отслеживать минимальные шаги работы интерпретатора и переходить в каждую реализуемую подпрограмму. После указанной смены режима нажимаем "Enter" дважды - сначала для прохождения через оператор "уровеньобращение(x7)", а затем - через оператор "равно(x18 ответзадачи(x17))". Просматриваем значение переменной x18 - найденный для подслучая ответ $\exists_n(x = (4n + 3)\pi/4 \ \& \ n$ – целое). Продолжаем трассировку по операторам программы, нажимая на клавишу "Enter". Через несколько шагов инициируется пустым словом накопитель x19. Как видно из следующего оператора, обращающегося к комментарию (выводимо $\ldots$), в x19 передается список всех использованных при решении задачи x17 посылок. Далее содержимое x19 присоединяется к накопителю x6, где создается общий список использованных приемом посылок. Следующий оператор - "замена(8 суффикс(x8 наборчленов(и x18)))" - заносит в накопитель x8 набор конъюнктивных членов ответа задачи x17. Таким образом, по окончании рассмотрения всех подслучаев длина набора x8 будет равна числу подслучаев, а его элементы будут определять найденные в подслучаях ответы. Еще одно нажатие клавиши "Enter" вызывает откат к оператору "позиция(x15 x14)", выбирающему следующий дизъюнктивный член. Чтобы сразу перейти от этой точки к моменту входа в решение вспомогательной задачи для очередного подслучая, нажимаем "Ctr-г", переводящее в оглавление программ. Здесь снова выбираем пункт "Рассмотрение подслучая для дизъюнктивного условия" и нажимаем "курсор вправо".

По окончании решения новой версии задачи x17 получаем ответ $\exists_n((x = (3n+1)\pi/3 \ \vee \ x = (3n+2)\pi/3) \ \& \ n -$ целое). Его вид подсказывает, что при решении задачи имел место еще один разбор случаев по дизъюнктивному условию. Чтобы посмотреть момент этого разбора случаев, можно было сразу же после выхода на контрольную точку "прием(3 3)", предшествующую обращению к задаче x17, выделить оператор "уровеньобращения(x7)" и нажать "Ctr-Enter". Такие действия привели бы к останову отладчика ЛОСа на выделенном операторе безотносительно к тому кадру, в котором была создана установка на прерывание. В нашей задаче это означало бы остановку при попытке нового разбора случаев внутри текущего подслучая. Напомним, что для выделения оператора в отладчике ЛОСа нужно нажать клавишу "курсор вниз", и далее использовать клавиши "курсор вправо - влево". Чтобы отменить выделение, нажимается "курсор вверх". Если после выделения оператора "уровеньобращения(x7)" была бы нажата клавиша "Enter", то остановка произошла бы только при обращении к данному оператору в том же самом кадре, т.е. для очередного подслучая текущей дизъюнкции.

Чтобы посмотреть действия приема по завершении рассмотрения подслучаев, выходим через пункт "Завершающая обработка ответов для подслучаев" оглавления программ на контрольную точку "прием(3 5)". Нажимая "2" для перехода в пооператорную трассировку, каждый очередной шаг выполняем нажатием клавиши "Enter". Доходим до оператора, присваивающего переменной x12 пересечение всех "известных" утверждений, входящих в ответы подслучаев. Далее переменной x13 присваивается результат занесения в конец списка x12 дизъюнкции конъюнкций оставшихся фрагментов ответов, а переменной x14 - конъюнкция утверждений списка x13. Она является результатом разбора случаев. Оператор "изменение(окрестность(x1 4)набор(x14))" заменяет весь список условий текущей задачи на одноэлементный список, состоящий из x14. Однако, в качестве ответа выдается не x14, а результат x15 упрощения его пакетным нормализатором "нормили". Этот нормализатор, в основном, реализован на ГЕНОЛОГе, и будет описан в последующих разделах. Продолжая трассировку, выполняем обращение к нормализатору и сравниваем утверждения x14, x15. Видно, что нормализатор объединил два параметрических описания в одно и вынес за скобку общее условие на параметр ($n -$ целое).

1.19.2 Прием редактирования параметрического описания

Следующий пример - прием редактирования параметрического описания. Он применяется к имеющему вид параметрического описания $\exists_{x_1 \ldots x_n} A$ условию задачи на описание, не разрешенному явно относительно неизвестной либо такому, что на определяемые в нем неизвестные существуют дополнительные внешние условия. Прием может быть найден в разделе "Приемы решателя" - "Общие приемы" - "Квантор существования" - "Обращение к задаче на редактирование параметрического описания". Контрольная точка выхода на прием через оглавление программ расположена в начале программы, и не каждое попадание на нее будет означать срабатывание приема. Чтобы найти задачи, в которых прием срабатывает, нужно ввести в его программу контрольную точку - оператор "трассировка(стоп 0)". Для этого выходим на программу приема через оглавление программ и нажимаем "курсор вниз". Переходим к следующему фрагменту, в начале которого расположен оператор "равно(x13 ответзадачи(x12))". Он осуществляет обращение к вспомогательной задаче на редактирование параметрического описания, так что контрольную точку удобно разместить непосредственно перед ним.

После того, как контрольная точка, вызывающая выход в отладчик ЛОСа при попытке применить прием, создана, можно выбрать какой-либо раздел задачника и запустить цикл решения задач. Учитывая то, что параметрические описания часто возникают в тригонометрических уравнениях и неравенствах, выберем подраздел задачника "Элементарная алгебра" - "Решение уравнений" - "Тригонометрические уравнения" - "Уравнения с неизвестной в знаменателе - 1" и запустим цикл решения нажатием клавиши "Ctr-з". Уже на третьей задаче будет выявлено срабатывание приема. Однако, оно относится к сравнительно простому случаю, и для дальнейшего перейдем сразу к 22-й задаче подраздела, где ситуация несколько более интересная. Чтобы перейти к данной задаче после появления кадра отладчика ЛОСа в цикле решения, нажимаем клавишу "Ctr-з". Лишь после этого нажатие "Esc" оборвет цикл и выведет в главное меню, откуда еще раз нужно будет зайти в задачник и выбрать указанную задачу. Она имеет следующее условие:

$$(\sec x)^2 - (\cos x + \sin x \operatorname{tg} \frac{x}{2}) = \frac{\sin(x - \frac{\pi}{6}) + \cos(\frac{\pi}{3} - x)}{\cos x}.$$

Запускаем решение задачи клавишей "о" и попадаем в кадр отладчика ЛОСа. Просматриваем текущее параметрическое описание, на котором сработал прием - его вхождение является значением переменной x2: $\exists_n(x = n\pi \ \& \ n - \text{целое})$. Просматриваем задачу x12 на редактирование данного параметрического описания. Так как ее решение еще не начато, используем сквозной просмотр набора x12, нажимая "K12". Пятый элемент этого набора - список условий; он состоит из утверждений $x = \pi n$, $n -$ целое, $\neg(\cos\frac{x}{2} = 0)$, $\neg(\cos x = 0)$, $x -$ число. Таким образом, смысл редактирования заключается в подстановке значения неизвестной x в сопровождающие условия и упрощении результатов. Заметим, что задача x12 имеет две неизвестные - x и n, причем неизвестная n объявлена несущественной. Для входа в трассировку по шагам решения задачи x12 нажимаем "Enter". Появляется первый оператор программы символа "описать". Чтобы инициировать указанную трассировку задачи, нажимаем клавишу пробела. Теперь каждое новое нажатие "Enter" будет приводить к очередному шагу решения. Начинаем трассировку. На первом шаге предпринимается исключение неизвестной x, выраженной через n. Возникает вспомогательная задача, имеющая условия "$n -$ целое, $\neg(\cos\frac{\pi n}{2} = 0)$, $\neg(\cos(\pi n) = 0)$, $\pi n -$ число" и неизвестную n, уже существенную. Следующее нажатие "Enter" - условие $\neg(\cos(\pi n) = 0)$ заменяется на константу "истина", которая отбрасывается приемом, не выводимым на уровень данной трассировки. Далее условие $\neg(\cos\frac{\pi n}{2} = 0)$ заменяется на

$$\neg(((-1)^{\frac{n}{2}} \text{ при } n - \text{even}, \text{ иначе } 0) = 0).$$

Последнее, после нескольких преобразований, дает условие $n - \text{even}$. Условие четности разворачивается в еще одно параметрическое описание - $\exists_m(n = 2m \ \& \ m - \text{целое})$.

Через несколько шагов общей стандартизации условий вновь срабатывает анализируемый нами прием, и мы опять оказываемся в кадре отладчика ЛОСа. На этот раз значением переменной x2 служит только что созданное параметрическое описание для n. Можно убедиться, что имеется внешний кадр программы того же приема. Для этого достаточно два раза нажать "Page Up". Вернувшись в текущий кадр (два нажатия "Page Down"), просматриваем задачу x12. Она имеет три условия - $n = 2m$, $m -$ целое, $n -$ целое. Входим в просмотр списка комментариев к задаче x12 (седьмой элемент набора x12 - список списков комментариев, и последний элемент данного списка - требуемый список). В нем обнаруживается комментарий (контекст A_1 A_2), где A_2 - отложенное до редактирования ответа равенство $x = \pi n$, с помощью которого выше была исключена переменная x. A_1 - та задача, в которой произошло данное исключение.

Продолжаем трассировку внутри новой задачи x12. Как и ранее, нажимаем "Enter", "пробел", и далее каждый новый шаг вызываем нажатием "Enter". Снова исключается неизвестная, явно выраженная через другую, на этот раз с помощью равенства $n = 2m$. Возникает задача с двумя условиями - "m – целое, $2m$ – целое". На первом шаге ее решения условие "$2m$ – целое" заменяется на константу "истина" и отбрасывается. В единственном оставшемся условии решатель усматривает ответ (относительно неизвестной m). Он обращается к процедуре "редакторответа", которая извлекает из комментария (контекст ...)отложенные условия $x = \pi n, n = 2m$, присоединяет их к текущему условию "m–целое", и обрабатывает весь этот список во вспомогательной задаче, имеющей цель "редакция". Неизвестными вспомогательной задачи служат переменные x, n, m, причем две последние - несущественные. Поэтому после подстановки значения $n = 2m$ в первое условие равенство для n отбрасывается. По завершении редактирования ответ приобретает вид $x = 2\pi m$ & m – целое. В момент его выдачи следует войти в кадр отладчика ЛОСа, нажав клавишу "ф". Это делается для того, чтобы продолжить трассировку применения анализируемого приема после решения задачи x12. Имеется в виду внутреннее срабатывание приема, относящееся к описанию $\exists_m(n = 2m$ & m–целое). Возвращаемся к кадру данного срабатывания приема, нажимая "Page Up". Войдя в него, нажимаем "2" для установки пооператорной трассировки в кадре. Далее нажимаем "Enter" и проходим оператор "равно(x13 ответзадачи(x12))". Убеждаемся в том, что значением переменной x13 служит найденный выше ответ.

Продолжаем трассировку в кадре. Через несколько шагов переменной x15 присваивается утверждение "$\exists_m(x = 2\pi m$ & m – целое)". Список условий текущей задачи (т.е. задачи на редактирование исходного параметрического описания) заменяется на одноэлементный набор, состоящий из указанного утверждения. Для получения окончательного ответа x16 к утверждению x15 применяется нормализатор общей стандартизации параметрических описаний "нормсуществует", который оставляет x15 неизменным. Наконец, реализуется оператор "ответ(x16)". Чтобы вернуться после выполнения данного оператора во внешний кадр, нужно переустановить режим трассировки на пошаговый (клавиша "1"). Если этого не сделать и продолжать нажимать "Enter", то следующее прерывание произойдет значительно позднее - снова по контрольной точке "трассировка(стоп 0)", но уже для другого подслучая. Переустанавливаем режим, нажимаем "Enter" и попадаем во внешний кадр на оператор "равно(x12 ответзадачи(x11))". Этот кадр - промежуточный на пути к тому кадру, где было начато редактирование исходного параметрического описания. Здесь реализуется прием, исключивший неизвестную x с помощью уравнения $x = n\pi$. Нажимаем "2" и продолжаем трассировку в кадре. Через несколько шагов приходим к оператору "ответ(x12)". Опять нажимаем "1", "Enter", и возвращаемся наконец к той точке, с которой было начато редактирование. Нажимаем "2", и далее, по уже известной траектории, доходим до оператора "ответ(x16)", завершающего срабатывание приема. Заметим, что утверждение "$\exists_m(x = 2\pi m$ & m – целое)"на последних шагах осталось неизменным - вся фактическая работа по редактированию ответа в рассматриваемом подслучае уже была выполнена оператором "редакторответа" до возвращения по цепочке кадров. В данной задаче прием срабатывает еще один раз, что легко проверить, отключив трассировку нажатием клавиши "0" и нажав "Enter". Снова произойдет прерывание по достижении контрольной точки "трассировка(стоп

0)". В качестве упражнения, рекомендуется самостоятельно проанализировать поведение решателя на этом срабатывании приема. По окончании нужно не забыть устранить из программы приема контрольную точку "трассировка(стоп 0)".

1.19.3 Прием исключения неизвестной задачи на описание, явно выраженной через остальные неизвестные

Часто применяется при решении задач с несколькими неизвестными прием, исключающий неизвестную x с помощью условия $x = t$, где выражение t не содержит x. Применение приема происходит безотносительно к тому, расположено ли x в левой или в правой части равенства. Чтобы он мог сработать, необходима предшествующая работа других приемов, создающих равенство указанного вида. С другой стороны, если такое равенство уже имеется, но применение приема нецелесообразно, могут понадобиться приемы, срабатывающие на уровне 0 и "маскирующие" явное выражение для x (например, перенесением всех ненулевых членов числового уравнения в одну часть).

Прием находится в разделе "Приемы решателя" - "Общие приемы" - "Равенство" - "Исключение неизвестной задачи на описание, явно выраженной через остальные неизвестные" оглавления программ. Для анализа его работы рассмотрим какую - либо систему алгебраических уравнений, например, задачу 8 из раздела задачника "Элементарная алгебра" - "Решение уравнений" - "Системы алгебраических уравнений - 2". Эта система имеет вид:

$$\left\{\begin{array}{rcl} x^3+y^3 & = & 7z^3 \\ x-y & = & 3z \\ y-z & = & x-2 \end{array}\right.$$

Для анализа поведения данного приема нет необходимости вставлять в его программу контрольную точку "трассировка(стоп 0)". Достаточно войти в просмотр задачи, нажать "л", выбрать концевой раздел оглавления программ, содержащий прием, и нажать клавишу "курсор вправо". Почти сразу возникает прерывание с выходом в отладчик ЛОСа. Вызываем на экран значение переменной x2 - текущее уравнение "$x = y + 3z$". Операторы текущего фрагмента программы подсказывают нам, что значением программной переменной x7 служит исключаемая неизвестная x, а значением переменной x6 - вхождение этой неизвестной в равенство. Чтобы определить общий контекст, выходим на уровень просмотра задачи (клавиша "з"). Остальные два условия имеют вид $-7z^3 + x^3 + y^3 = 0$, $y - z = x - 2$. Возвращаемся в кадр отладчика (клавиша "ф"), и для продолжения трассировки в этом кадре нажимаем "2". После нескольких шагов, связанных с обработкой фильтров и учетом специальных случаев, попадаем на операторы "операнд(x2 x8) не(равно(x6 x8)) равно(x9 подтерм(x8))". Они выделяют противоположную часть x8 рассматриваемого равенства; переменной x9 присваивается выражение $y + 3z$, расположенное в этой части. Продолжаем трассировку до оператора, присваивающего переменной x10 набор результатов подстановки выражения $y + 3z$ вместо x в остальные условия. Нажимаем "K10" для просмотра этого набора. Далее создается вспомогательная задача x11, имеющая x10 своим списком условий. Просматриваем ее, нажимая "K11". В списке целей (8 - й элемент набора задачи) находим измененную цель (неизвестные y z), в которой отсутствует исключенная неизвестная x.

Через несколько шагов появляется оператор "замечание(x11 обращение)", сопровождающий новую задачу комментарием "обращение". Этот комментарий необходим для того, чтобы в процессе трассировки решения "по срабатываниям приемов" система автоматически входила в просмотр решения вспомогательной задачи x11. Оно существенно важно для понимания общего процесса решения, и пропуск его был бы нежелателен.

Еще одна важная подробность - при продолжении трассировки задача x11 сопровождается комментарием (контекст x1 A), где A - одноэлементный набор, состоящий из равенства $x = y + 3z$.

Наконец, коррекция новой задачи x11 завершается, и возникает оператор "равно(x12 ответзадачи(x11))". При рассмотрении предыдущего примера мы уже оказывались в его окрестности. В качестве самостоятельного упражнения, можно продолжить трассировку процесса решения задачи x11 до получения ответа и момента использования комментария (контекст ...). Пока мы пропускаем эту трассировку, нажимая "Enter" в очередной раз. Выполнение оператора "равно(x12 ответзадачи(x11))" требует определенного времени, так как фактически в нем решение всей задачи доводится до конца. Получаем результат x12 и выводим его на экран: $z = 1/2 \: \& \: x = 1 \: \& \: y = -1/2$. После нескольких дальнейших шагов сопровождающего характера (например, коррекции комментария (выводимо ...) для учета использованных посылок) прием выдает ответ x12.

1.19.4 Попытка решения уравнения из блока анализа, имеющего единственную неизвестную

В заключение рассмотрим прием, усматривающий в блоке анализа уравнение с единственной числовой неизвестной x и предпринимающий попытку решить его относительно x. Заметим, что к уравнениям относительно единственного атомарного числового подвыражения t с неизвестными, не являющегося переменной, данный прием неприменим. Чтобы разрешить такие уравнения относительно выражения t, нужно вводить для этого выражения вспомогательное обозначение - новую неизвестную y, и лишь тогда прием сработает.

Найти программу приема можно в разделе "Приемы решателя" - "Общие приемы" - "Равенство" - "Попытка решения уравнения из блока анализа, имеющего единственную неизвестную". Он часто срабатывает в тех предметных областях, где интенсивно используется блок анализа, например, в геометрии. Рассмотрим в качестве примера простую планиметрическюу задачу на вычисление, расположенную в разделе задачника "Элементарная геометрия" - "Задачи на вычисление" - "Треугольник" - "Общий случай" - "Биссектрисы" и имеющую номер 1. Дан треугольник ABC, у которого известны длины b, c сторон AC, AB, причем длина биссектрисы AD равна длине отрезка DB. Требуется найти длину стороны BC.

Чтобы проанализировать срабатывание приема, войдем в просмотр условия задачи, нажатием клавиши "л" перейдем в оглавление программ, найдем указанный выше подраздел приема, выберем в нем концевой пункт "Формирование вспомогательной задачи на описание x9", и нажмем "Enter". Почти сразу появится кадр отладчика - контрольная точка "прием(1 3)", после которой расположен выделенный малиновым цветом оператор "равно(x9 ...)". Как следует из названия пункта, данный оператор присваивает переменной x9 заготовку создаваемой вспомогательной задачи на решение уравнения. Просматриваем само уравнение - его корневое вхождение является значением переменной x2. Уравнение имеет вид "$(b+x)(x-b) = bc$".

В качестве самостоятельного упражнения, можно порекомендовать просмотреть предысторию появления данного уравнения путем трассировки на уровне применений приемов. При этом нужно постараться точно определить момент смены режима трассировки, чтобы снова попасть в данный кадр отладчика ЛОСа.

Нажатием клавиши "2" входим в режим пооператорной трассировки, нажимаем "Enter" и просматриваем с помощью процедуры сквозного просмотра заготовку задачи x9 (нажимается "K9"). Видно, что в список посылок отобраны утверждения $0 < c$, $0 < b$, b – число, c – число. Единственным пока условием является уравнение; неизвестной служит переменная x. Вернувшись из сквозного просмотра в кадр отладчика, продолжаем нажимать "Enter". Попадаем в цикл просмотра условий задачи x9 и обращений к справочнику "одз", пополняющему список условий сопровождающими утверждениями. На некотором шаге этого цикла переменной x16 присваивается утверждение "x - число", заносимое в список x10. Он представляет собой накопитель условий и посылок, которые впоследствии будут присоединены к задаче x9. После цикла учета о.д.з. продолжаются попытки пополнения списка условий задачи x9 путем усмотрения различных простых дополнительных ограничений на x, вытекающих из всего списка посылок блока анализа. На некотором шаге трассировки обнаруживаем присвоение переменной x18 утверждения $0 \leq x$, вытекающего из равенства $l(BC) = x$. Далее предпринимается обращение к оператору "разныеточки", устанавливающему различие точек B, C, и замена x18 на более сильное утверждение $0 < x$. Затем x18 присоединяется к списку x10. Дальнейшая трассировка не изменяет содержащих x утверждений списка x10, хотя в этот список попадают еще два утверждения - $0 < b$ и $0 < c$. После достаточно длинной цепочки шагов (в том числе цикла попыток упрощения утверждений из x10 относительно списка посылок задачи x9), приходим к оператору "длялюбого(x12 если входит(x12 x10)то альтернатива(...))". Этот оператор добавляет к списку условий задачи x9 все утверждения списка x10, содержащие x, а к списку посылок - все остальные утверждения. После его выполнения идут операторы "уровеньобращения(6) равно(x12 ответзадачи(x9))", обращающиеся к решению задачи x9. Здесь можно снова нажать "K9" и посмотреть на окончательную версию данной задачи. Если нужно посмотреть ход решения задачи x9, то, вернувшись в кадр отладчика, устанавливаем пошаговый режим трассировки (клавиша "1"), нажимаем "Enter" и оказываемся в программе символа "описать". Нажимаем "пробел" для перехода к трассировке по срабатываниям приемов, и далее прослеживаем ход решения. Получив уравнение $x^2 = bc+b^2$, решатель сразу извлекает корень, отбрасывая отрицательное значение. Здесь помогает извлеченное из блока анализа дополнительное условие $0 < x$. После упрощения, выдается ответ $x = \sqrt{b(b+c)}$. Для возвращения в отладчик ЛОСа нажимаем "ф". Нажимая "PageUp", возвращаемся в тот кадр, где был расположен оператор "равно(x12 ответзадачи(x9))". Здесь переходим в режим трассировки по операторам данного кадра (клавиша "2"). Нажимая далее "Enter", выходим на оператор, следующий за "равно(x12 ответзадачи(x9))". Убеждаемся, что значением переменной x12 стал найденный выше ответ. После выполнения ряда операторов, связанных с коррекцией комментариев для предстоящего преобразования, приходим к оператору "удалениепосылки(x1 x3)". Он удаляет посылку - рассматриваемое уравнение блока анализа. Вместо этого в список посылок вводятся конъюнктивные члены утверждения x12, сопровождаемые необходимыми комментариями. Выполнение приема завершает оператор "пересмотр".

1.19.5 Упражнения

1. Найти в разделе задачника "Теория множеств" задачу, использующую прием доказательства от противного. Посмотреть ход рассуждений при получении

противоречия.

2. Найти в том же разделе все задачи, использующие прием подбора подстановки вместо неизвестных, преобразующей все условия в посылки.

3. Найти прием "фиксация значения связанной переменной через равенство в антецеденте". Найти в разделе "Элементарная алгебра" задачу, в которой он применяется.

4. В подразделе оглавления программ "Импликативное условие задачи на описание, не требующей полного ответа" найти пункт "Попытка обратного вывода с помощью кванторной посылки". Найти в разделе задачника "Математический анализ" - "Задачи на доказательство" задачу, в которой этот прием срабатывает. Проанализировать это срабатывание.

5. Найти прием "Решение относительно параметра числового равенства без неизвестных". Найти в разделе задачника "Логарифмические уравнения" такую задачу, где этот прием применяется к равенству, не являющемуся линейным относительно своего параметра.

6. Найти прием "Подбор примера с заменой функциональной неизвестной на обычную". Найти в разделе задачника "Пределы" - "Теоретические задачи" задачу, в которой этот прием срабатывает. Проанализировать срабатывание.

Глава 2

Общие процедуры, используемые в приемах

Приемы решателя используют большое количество вспомогательных процедур, реализованных на ЛОСе. Иногда трассировка процессов решения задач требует понимания того, как устроены эти процедуры. Кроме того, во многих случаях обучение решателя может потребовать пополнения списка выполняемых ими действий. По существу, в этих процедурах сосредоточена "автоматика нижнего уровня" решателя, обеспечивающая необходимую коррекцию разнообразных структур данных, отображающих текущую ситуацию. Приводимое ниже краткое описание устройства наиболее важных таких процедур дает минимальную необходимую в обоих случаях информацию. При чтении данного раздела рекомендуется входить в соответствующие ЛОС - программы и находить те их точки, о которых говорится в тексте.

2.1 Преобразования задач

Начнем с вспомогательных процедур, преобразующих задачу. Их количество не очень велико, но практически все изменения задач, выполняемые приемами, проходят через эти процедуры. Таким образом обеспечивается возможность перехвата и блокировки преобразования уже после того, как прием принял решение о его выполнении. Необходимость блокировки может быть вызвана какими - либо общими соображениями, которые проще рассматривать в одной точке, чем дублировать их рассмотрение во многих программах. Кроме того, блокировка может быть вызвана особенностями трассировки, наличием режима вывода в базе теорем, потребностями генератора приемов и т.п. Обычно процедура, выполняющая преобразование задачи, должна обеспечить коррекцию многих сопутствующих структур данных, в основном, локализованных в комментариях задачи. Иногда она может даже обращаться к вспомогательным процессам логического уровня, вызывающим дополнительные изменения посылок и условий задачи.

2.1.1 Процедура "заменавхождения"

Это наиболее часто используемая процедура приемов, выполняющая замену подтерма по заданному вхождению в задачу. Формат обращения к ней имеет вид "заменавхождения(x1 x2 x3 x4 x5 x6)", где x1, x2, x3 - координата вхождения в задачу x4 заменяемого подтерма; x5 - заменяющий терм либо символ; x6 - набор информационных элементов, дополняющих и уточняющих реализацию процедуры. Напомним, что координатой вхождения в задачу является тройка (A_1, A_2, A_3), где A_1 - вхождение рассматриваемого подтерма в некоторый терм задачи (т.е. в ее посылку либо условие); A_2 - вхождение данного терма задачи в его внешний список (т.е. список условий, посылок либо в задачу); A_3 - указатель условия (1) либо посылки (0).

В оглавлении программ процедуре отведен подраздел "Приемы решателя" - "Общие процедуры, используемые в приемах" - "Процедура ЗАМЕНАВХОЖДЕНИЯ". Выбираем в этом подразделе пункт "Исходная точка", нажимаем "курсор вправо", и попадаем на начальный фрагмент программы.

Прежде всего, проверяется, что заменяемый и заменяющий термы различны. Если это не так, то - выход по истинностному значению "ложь". Иначе - переход через "ветвь 2".

Если заменяющий терм представляет собой символ (логический либо переменной), то он преобразуется в формат терма. Затем - переход через "ветвь 1".

Следующий фрагмент относится к редкой ситуации, встречающейся при выводе теорем, и мы его пропускаем, переходя через "ветвь 1".

Здесь начинается ветвь проверки условий, блокирующих применение преобразования. Прежде всего, анализируется наличие комментария (приемы A_1 A_2 A_3) к посылкам исходной задачи, блокирующего применение приема ГЕНОЛОГа, адресуемого тройкой (A_1, A_2, A_3). Напомним, что элементы этой тройки суть, соответственно, логический символ, номер узла этого символа в базе приемов, к которому отнесен прием, и заголовок приема. Такой комментарий мог быть создан, чтобы временно заблокировать любые срабатывания данного приема (например, при работе системы вывода теорем или при отладке). Если A_3 в комментарии равно 0, то блокируются все приемы, относящиеся к тому же самому узлу. При наличии в x6 информационного элемента (прием ...), адресующего текущий выполняемый прием ГЕНОЛОГа, этот адрес сравнивается с указанным в комментарии, и при совпадении (с точностью до случая $A_3 = 0$) прием блокируется.

Далее проверяется наличие комментария (вычерк ...), который аналогичен комментарию (приемы ...), но вводится в других ситуациях - в циклах тестирования избыточности применений приемов вывода либо замены.

После рассмотрения указанных комментариев - откат к переходу через "ветвь 1", расположенному перед оператором "исходнаязадача(x7)". Здесь продолжается перечисление редких ситуаций, в которых применение приема блокируется. Это происходит, если решается задача на исследование, а преобразуемая посылка - исходная для нее и помечена комментарием "пассив". Другой случай - решение задачи на предварительный анализ текста, если пытается сработать прием, анализирующий типы объектов. Для этого приема нужен корректный логический контекст, который еще не создан.

Снова переходим через "ветвь 1". В этом фрагменте, начинающемся с операторов "ветвь 1 ветвь 2 заголовок(x5 ложь)", происходит учет сопровождения по о.д.з. Если предполагаемая замена относится к утверждению, которое используется как сопровождающее по о.д.з. для какого-то другого терма задачи, то она обычно блокируется. Это делается потому, что иначе усмотрение принадлежности области допустимых значений каких-то подвыражений задачи могло бы оказаться затрудненным. Такую ситуацию будем характеризовать как нарушение структуры сопровождения по о.д.з. Она крайне нежелательна, так как может приводить к блокировке срабатываний самых простых и естественных приемов. Фрагменты, обеспечивающие контроль сохранения структуры сопровождения, достижимы через оператор "ветвь 2". До перехода к ним анализируются случаи, когда заменяющий терм является логической константой. Если он является константой "ложь" и заменяемое вхождение - корневое либо расположено под импликацией или конъюнкцией, то весь логический контекст, к которому относится преобразуемое утверждение, становится ложным. Поэтому сохранять в нем структуру сопровождения по о.д.з. нет необходимости, и переход через "ветвь 2" отменяется. То же самое выполняется, если заменяющее утверждение - константа "истина", а заменяемое расположено под отрицанием, т.е. после замены снова появляется константа "ложь" в одном из указанных выше контекстов.

Если переход через "ветвь 2" все же реализуется, то прежде всего проверяется отсутствие в x6 информационного элемента "результподст". Он означает, что замена представляет собой стандартизацию обозначений с помощью равенства $t_1 = t_2$, расположенного в контексте заменяемого терма. Здесь блокировки из соображений сохранения сопровождения по о.д.з. отменяются. Обычно данное преобразование, применяемое как к сопровождающему, так и сопровождаемому термам, сохраняет структуру сопровождения.

Для дальнейшего напомним, что информация о сопровождении по о.д.з. создается в начале решения задачи в комментариях (сопровождение A). Здесь A - набор пар $(B_1\ B_2)$, таких, что B_1 - список утверждений задачи (посылок, условий, подтермов конъюнкций и импликаций, и т.п.), из которых выводится совокупность условий на о.д.з. для терма B_2. В процессе решения задачи данные комментарии постоянно пополняются и корректируются процедурами, изменяющими задачу. В частности, это делается и описываемой процедурой. На данном, начальном этапе, речь идет не о коррекциях комментариев (сопровождение ...), а лишь об использовании их для блокировки замены.

Если в x6 имеется информационный элемент "сопровождение", то прием, выполняющий замену, берет на себя всю ответственность за возможное нарушение структуры сопровождения. Обычно это делают приемы, преобразования которых синхронно изменяют и сопровождающий, и сопровождаемый термы. Тогда блокировки не происходит, но во всех ссылках на преобразуемое утверждение из комментария (сопровождение ...) выполняется изменение его на преобразованную версию. Последняя создается здесь же, задолго до выполнения данной процедурой фактической замены утверждения (там она будет создана повторно).

Если элемента "сопровождение" в x6 нет, то применяется процедура "сопровождение(x4 x1 x2 x3)", которая определяет с помощью комментариев (сопровождение ...), используется ли преобразуемое утверждение для сопровождения по о.д.з. Если используется, причем имеет хотя бы одну переменную, то замена блокируется.

В случае, когда замена не была заблокирована, происходит откат к переходу через "ветвь 1". Данный фрагмент программы начинается с операторов "ветвь 1 тип(x4 описать) равно(x3 1) цель(x4 редуцирование)". Цель "редуцирование" обычно возникает в задачах, связанных с логическим выводом теорем. Поэтому мы пока пропустим фрагмент и перейдем к следующему фрагменту через оператор "ветвь 1".

После контрольной точки "прием(63)" располагается фрагмент, обеспечивающий контроль зацикливаний. Чтобы этот контроль выполнялся, необходимо наличие элемента "повторение" в списке x6 опций обращения к процедуре. Для контроля используется комментарий (повторение A) к преобразуемому терму задачи (в случае условия задачи на доказательство либо преобразование - комментарий к задаче). Набор A этого комментария хранит пары $(t\ N)$, где t - терм, на который ранее заменялся данный терм задачи; N - сколько раз предпринималась замена. Как только N становится равно 4 (считая текущую попытку), замена блокируется. Список A доводится до длины 12, и далее при каждом занесении в него новой пары отбрасывается самая "старая" пара (она идет в начале списка). Таким образом легко обнаруживать большинство типичных зацикливаний, возникающих при работе рещателя. Более того, они автоматически обрываются - достаточно лишь, чтобы те приемы, которые априори могут привести к зацикливанию, вводили элемент "повторение" в обращение к процедуре "заменавхождения".

Продолжая перемещение по цепочке операторов "ветвь 1", переходим к следующему фрагменту, содержащему контрольную точку "прием(55)". Здесь предпринимается расчистка буфера обращений к нормализаторам, определяемая информационным элементом (контрольнормализации S T) набора x6. Напомним, что роль буфера обращений к нормализатору с заголовком S играет комментарий (контрольнормализации S A v). У него A - набор наборов (1 t P K r Q), где t - преобразуемый терм; P - список посылок обращения; K - список входных комментариев обращения, от которых может зависеть результат; r - результат преобразований терма t; Q - список использованных посылок. Информационный элемент (контрольнормализации S T) определяет отбрасывание из буфера всех наборов, у которых r содержит подтерм T. Используется данная возможность крайне редко - она предотвращает в ряде случаев нежелательные преобразования с использованием готовых результатов, сохраненных в буфере.

Еще один переход через "ветвь 1" приводит к фрагменту с оператором "ключ(комментариипосылок(x7)трассперечисл x8)". Здесь проверяется наличие комментария (трассперечисл A_1 A_2 A_3) к исходной задаче. Он служит как указатель на вход в трассировку по шагам решения текущей задачи при попытке применения приема ГЕНОЛОГа, имеющего адрес (A_1, A_2, A_3). Процедура сравнивает применяемый прием с приемом, заданным в комментарии, и при совпадении инициирует указанный режим трассировки.

Далее через оператор "ветвь 2" попадаем в фрагмент с контрольной точкой "прием(7)". Здесь выполняется обращение к процедурам "учетприменения", "символы", которые регистрируют факт применения в данной задаче данного приема. Эта информация потом будет сохранена в архиве задачника, чтобы можно было находить по ней все задачи, где прием срабатывал. Пока она сохраняется в комментарии (задача A_1 A_2) к исходной задаче, при условии, что этот комментарий ранее был создан. В списке A_1 перечисляются все логические символы, возникающие при решении задачи в посылках и условиях; в списке A_2 - ссылки на сработавшие приемы. Они группируются по заголовкам, так что каждый элемент списка A_2 есть пара (B_1 B_2), где B_1 - заголовок приема; B_2 - набор пар (логический символ - набор номеров узлов приемов) для приемов с данным заголовком. Кроме того, процедура "учетприменения" сохраняет в комментарии (выписка $\ldots$) информацию о холостом ходе приемов и числе их срабатываний. Такой комментарий вводится при запуске серии решения задач клавишей "Ctr-x".

После обращения к процедуре "учетприменения" - откат к переходу "ветвь 1". Новый фрагмент начинается с операторов "равно(x7 пустоеслово) равно(x8 0)равно(x9 0)". Он определяет информацию, необходимую для коррекции сопровождения по о.д.з. при выполняемой замене. Значением переменной x7 станет список пар (U t), где t - неодносимвольный подтерм заменяющего терма, не встречающийся в заменяемом терме; U - список утверждений, использованных для усмотрения условий на о.д.з. для t. Эти утверждения берутся из нескольких различных источников - из контекста заменяемого терма; из контекста вхождения t в заменяющий терм; из числа утверждений F, выделенных имеющимися в x6 информационными элементами (вывод F $\ldots$); из числа утверждений F выделенных имеющимися в x6 информационными элементами (коррекцияпосылок $F\ldots$). Программа просматривает подтермы t заменяющего терма, находит условия на о.д.з. для них, и пытается усмотреть эти условия из указанных источников. Там, где это удается, список U пополняется использованными утверждениями.

Значением x8 станет список всех посылок и условий задачи, относящихся к контексту заменяемого терма задачи; значением x9 - список вообще всех утверждений, относящихся к этому контексту. Переменной x10 присваивается заменяемый терм. Заметим, что в тех случаях, когда учет о.д.з. для заменяющего терма не нужен (например, в задачах на анализ текста либо при завершающем редактировании ответа), значения x7,x8,x9 остаются неизменными, т.е. равными пустому слову и нулю.

Переходим через "ветвь 1" и оказываемся в фрагменте, начинающемся с контрольной точки "прием(9)". Если заменяемое вхождение не является корневым, а расположено внутри каких-либо логических контекстов (конъюнкция, кванторная импликация, описатель "отображение"), то при замене может понадобиться пополнение этих контекстов утверждениями, необходимыми для сопровождения по о.д.з. заменяющего терма. Разумеется, никакой новой информации такие утверждения не несут - они являются следствиями уже имеющихся, но явное их указание необходимо, чтобы не было трудностей с проверками на о.д.з. при реализации последующих приемов. С учетом такого пополнения, замена будет затрагивать не указываемый в обращении к процедуре "заменавхождения" терм, а некоторый его надтерм, включающий все пополняемые логические контексты. Для составления полного "плана" замен вводится набор x11 $= (K_1\ldots K_m)$ троек K_i, обозначающих последовательные вложенные друг в друга логические контексты, к которым относится заменяемый терм. Первым элементом тройки является вхождение заменяемого терма при $i = 1$ либо логического контекста при $i > 1$; вторым - список утверждений этого контекста, относящихся к области заменяемого терма; третьим - заменяющий терм при $i = 1$ и список дополнительных утверждений при $i > 1$. Рассматриваются следующие типы логических контекстов:

1. Заменяемое вхождение расположено внутри некоторого операнда конъюнкции. Тогда к логическому контексту относятся все прочие операнды этой конъюнкции;
2. Заменяемое вхождение расположено внутри некоторого антецедента либо консеквента кванторной импликации. Тогда к логическому контексту относятся все антецеденты, не содержащие данного вхождения;
3. Заменяемое вхождение расположено внутри выражения t описателя "отображение($x_1\ldots x_n$ P t)", причем утверждение P не имеет заголовка "и". Тогда P относится к логическому контексту.

После составления списка x11 - переход через "ветвь 1" к фрагменту, начинающемуся с оператора "ключ(x6 коррекцияпосылок x12)". Здесь предпринимается рассмотрение содержащегося в x6 информационного элемента (коррекцияпосылок A). Такой элемент создается приемом, чтобы перечислить утверждения, которые могут понадобиться для сопровождения по о.д.з. заменяющего терма, и указать их обоснования. В особенности он важен, если прием обращался к вспомогательным задачам либо трудоемким пакетным операторам, и заменяющий терм возник после длинной цепочки преобразований. Данный элемент создается приемом в начале его работы, и далее передается всем вспомогательным процессам, чтобы они могли занести в него информацию о выводе сопровождающих условий для новых термов. Набор A состоит из троек $(B_1\ B_2\ B_3)$, где B_1 - сопровождающее утверждение; B_3 - список утверждений, являющихся обоснованием утверждения B_1; B_2 - 0 либо такое подмножество утверждений списка B_3, которое является следствием B_1 и остальных утверждений из B_3. Его элементы, после регистрации утверждения B_1, могут быть удалены из задачи без потери информации.

Если предпринимается завершающая обработка ответа, то x6 содержит элемент "свертка", указывающий, что имеет место ослабленный контроль за сохранением структуры сопровождения по о.д.з. Это - вынужденная мера, так как сохранение указанной структуры может входить в противоречие с требованиями на стандартный вид ответа.

При отсутствии элемента "свертка" просматриваются такие тройки $(B_1\ B_2\ B_3)$ из набора A, что утверждение B_1 либо представляет собой константу "ложь", либо имеет общую свободную переменную с заменяющим термом. Находится список x14 конъюнктивных членов утверждения B_1. Из него отбирается подсписок x15 утверждений, которые фактически будут регистрироваться в контексте замены. Если x6 имеет элемент "новаяпосылка", то x15 совпадает с x14. В противном случае к x15 относятся лишь элементы списка x14, упомянутые в списке x7 как представляющие интерес для сопровождения по о.д.з. заменяющего терма.

Чтобы определить точку регистрации утверждений из x15, вводится список x16. При наличии в x6 элемента "новаяпосылка" он пуст, иначе - состоит из всех неконстантных утверждений списка B_3. Некоторые утверждения списка x16 могут отсутствовать в контексте замены, но представлять собой элемент C_1 другой тройки $(C_1\ C_2\ C_3)$ набора A. Они изымаются из x16, и вместо них заносятся все утверждения списка C_3. После такого преобразования списка x16 начинается выбор точки регистрации утверждений списка x15. Если x16 содержится в списке посылок задачи, то регистрация происходит в списке посылок. Если заменяемое вхождение само относилось к некоторой посылке P, то определяется новое вхождение P в измененный список посылок, и корректируется значение переменной x2. Если x16 содержится в списке условий задачи на описание, то утверждения из x15 заносятся в список условий. Как и выше, корректируется x2. В остальных случаях просматривается список логических контекстов x11. Как только находится первый (ближайший к заменяемому вхождению) логический контекст, пересекающийся с x16, так все отличные от посылок и условий задачи утверждения списка x15, не представленные в найденном контексте, регистрируются как добавляемые к нему.

Если набор x6 имел элемент "свертка", то извлекаемая из A информация о выводимости утверждений F, используемых для сопровождения по о.д.з., лишь регистрируется в комментарии (стандследствие S). Здесь S - набор пар $(T\ F)$, где T - список утверждений, являющихся обоснованием утверждения F. В списки посылок и условий утверждения F не заносятся.

После указанной обработки элементов (коррекцияпосылок ...) - откат к переходу через "ветвь 1". Появляется фрагмент, начинающийся с оператора "ключ(x6 вывод x12)". Здесь анализируется принадлежащий набору x6 информационный элемент (вывод $A_1\ A_2\ A_3$). Он определяет утверждение A_1, которое следует занести в контекст заменяемого терма. Такие утверждения задаются соответствующими указателями приема. Если $A_2 = 0$, то утверждение присоединяется к посылкам задачи либо к внешнему логическому контексту заменяемого терма; в противном случае оно может быть занесено и в список условий. A_3 - набор элементов (выводимо ...), определяющих необходимое для обоснования A_1 дополнение к общему списку использованных приемом утверждений. В наборе A_3 может также находиться элемент (примечание ...), перечисляющий комментарии, которыми нужно сопроводить утверждение A_1. Возможен случай $A_1 = 0$, когда никаких действий не предпринимается. Этот случай возникает, если A_1 определяется условным выражением, проверяющим целесообразность дополнительного вывода.

Для принятия решения о точке регистрации конъюнктивных членов утверждения A_1 составляется расширенный список x14 утверждений, использованных приемом. Если x14 содержится в списке посылок, то регистрация происходит в списке посылок. Если он содержится в общем контексте замены, $A_2 \neq 0$, а задача имеет тип "описать", то регистрация происходит в списке условий. Если x14 пересекается с логическим контекстом заменяемого терма, то берется ближайший к нему логический контекст, и регистрация предпринимается в нем. В прочих случаях регистрация A_1 происходит в списке посылок.

После обработки элементов (вывод ...) - переход через "ветвь 1". Проверяется, имеется ли комментарий (сопровождение A_1), определяющий структуру сопровождения по о.д.з. Если такого комментария нет, то он вводится, с пустым списком A_1. Далее - переход через "ветвь 1".

Переменной x12 присваивается список всех утверждений, использованных приемом для обоснования замены. Они извлекаются из информационных элементов (выводимо ...), находящихся в списке x6. Переменной x13 присваивается комментарий (сопровождение A); набор A этого комментария присваивается переменной x14. Напомним, что A состоит из пар $(P,\ t)$, где t - выражение; P - список встречающихся в задаче утверждений, из которых вытекают условия на о.д.з. для t.

После контрольной точки "прием(13)" начинается цикл исключения утверждений, которые далее в задаче не нужны - они составляли сопровождение по о.д.з. для подтермов заменяемого терма, исчезающих при замене, а сами являлись следствиями остающихся утверждений. Просматриваются такие пары $(P,\ t)$ списка x14, для которых имеется вхождение терма t в заменяемый терм и нет вхождения t в заменяющий терм. Проверяется, что заменяемый терм не является утверждением, содержащимся в P. Далее просматриваются утверждения F списка P, не являющиеся согласно списку x7 элементами обоснования о.д.з. для заменяющего терма. Проверяется, что F не входит в такие пары $(P',\ t')$ списка x14, для которых t' - подтерм заменяющего терма. Далее рассматриваются следующие случаи:

1. F - посылка задачи. По списку x14 проверяется, что F используется для сопровождения по о.д.з только таких термов, которые расположены внутри заменяемого терма. Анализируется элемент (коррекцияпосылок ...) списка x6. Если из него усматривается, что F является следствием нового утверждения, используемого для сопровождения по о.д.з., а также некоторых других посылок задачи, то оно исключается из списка посылок. В случае задачи на исследование дополнительно к этому предпринимается попытка быстрого усмотрения истинности F из других посылок задачи. Здесь используются проверочные операторы. При успехе F удаляется.

2. F - условие задачи на описание. Действия аналогичны предыдущему пункту, однако с помощью элемента (коррекцияпосылок ...) проверяется, что F является следствием нового утверждения, а также других условий и посылок задачи. Дополнительная проверка с помощью проверочного оператора здесь происходит без ограничений на трудоемкость.

3. F содержится в логическом контексте K заменяемого терма (см. список x11). По списку x14 проверяется, что F используется для сопровождения по о.д.з только таких термов, которые расположены внутри заменяемого терма. Переменной x9 присваивается список утверждений, относящихся к контексту логического контекста K (включая K). Далее снова предпринимается попытка усмотреть, что F - следствие всех прочих утверждений списка x9. Сначала это делается с помощью комментариев (коррекцияпосылок ...), а затем с помощью проверочных операторов. При успехе F исключается из списка утверждений логического контекста K, находящегося в структуре данных x11.

По окончании попыток удаления не используемых далее утверждений - откат к переходу через "ветвь 1". Возникает фрагмент, начинающийся с контрольной точки "прием(14)". Здесь происходит анализ надтермов заменяемого терма. Они, как и сам заменяемый терм, требуют коррекции структуры данных сопровождения по о.д.з., в том числе ввода дополнительных посылок и условий, если это необходимо. Переменная x15 последовательно принимает в качестве своих значений вхождения указанных надтермов.

Прежде всего, проверяется, не является ли x15 вхождением одного из логических контекстов, упомянутых в списке x11. Это происходит после контрольной точки "прием(32)". Если является, то на последнюю позицию тройки списка x11, представляющей контекст x15, заносится утверждение для той версии контекста, которая возникнет после замены. Здесь учитываются как добавляемые утверждения, так и вид заменяющего терма.

Далее - переход через "ветвь 2" перед контрольной точкой "прием(32)", где начинается собственно анализ сопровождения по о.д.з. Последовательно просматриваются пары $(P\ t)$ списка x14, указывающие группы P утверждений, используемых для обоснования сопровождения по о.д.з. терма t. Находится такая пара x16, у которой t - текущий надтерм x15. Чтобы определить вид терма t', в который t перейдет после замены, просматривается список x11 и находится наибольший логический контекст Q, содержащийся в x15 (т.е. в t). При отсутствии такового берется сам заменяемый терм. Определяется результат x19 замены в текущем терме задачи T (посылке либо условии) этого контекста на его новую версию Q', найденную в том же цикле раньше. В x19 определяется вхождение x20, соответствующее вхождению x15 в T. Теперь это - вхождение измененного терма t'. С помощью справочника "одз" находится список утверждений, необходимых для сопровождения терма t'. В нем выделяется подсписок x22 неконстантных утверждений, отличных от простейших указателей типа объектов.

После контрольной точки "прием(45)" выполняется попытка скорректировать пару $(P\ t)$ за счет усмотрения заменяемого терма в утверждениях списка P и реализации в них синхронной замены. Сначала проверяется, не совпадает ли P со списком x22. Если совпадает, то коррекция пары сводится к замене в ней t на t'. Иначе - переход через "ветвь 3". Здесь значением переменной x23 является список P; значением переменной x24 - подтерм Q (выделенный выше логический контекст либо заменяемый терм). Переменной x25 присваивается копия списка P, в которую будут вноситься изменения для обеспечения сопровождения t'. Элементы списка x25 просматриваются, и проверяется, что те из них, которые имеют подтерм x24 (в явном виде либо, при ассоциативно - коммутативном заголовке, как подмножество операндов), не имеют двух различных его вхождений. Если данное условие выполнено, то в этих элементах реализуется замена Q на Q'. По окончании цикла просмотра списка x25 - переход через "иначе 1".

Здесь проверяется, что образующие сопровождение терма t' утверждения списка x22 усматриваются с помощью проверочных операторов из утверждений списка x25. Затем начинается цикл просмотра тех утверждений F списка x23, для которых соответствующее утверждение F' списка x25 получено заменой подтерма Q на Q'. Выделяются следующие подслучаи:

1. F - посылка задачи (контрольная точка "прием(46)"). Если удается установить, что F далее не нужна, то она заменяется на F', причем сразу корректируется пара $(P,\ t)$. В противном случае (переход через "ветвь 3") выводится новая посылка F', обоснованием которой служат F и все использованные приемом утверждения.
2. F - условие задачи (контрольная точка "прием(47)"). Действия аналогичны предыдущему пункту, но заменяется либо добавляется условие задачи.
3. F - утверждение логического контекста, внутри которого расположен контекст Q (контрольная точка "прием(48)"). Действия аналогичны предыдущему: если F далее не нужно, то оно заменяется в рассматриваемом контексте на F', иначе F' добавляется к контексту.

В тех случаях, когда происходит замена F на F', выполняется коррекция пары $(P,\ t)$. При этом индикатор коррекции x28 устанавливался на 1. Если по окончании цикла просмотра F, F' он остался равен 0, то список x7 пополняется парой (x25, t').

Если попытка скорректировать "старое" сопровождение для t, выполнявшаяся после контрольной точки "прием(45)", не удалась, то переход через оператор "ветвь 2" перед данной контрольной точкой. Здесь сопровождение для t' будет создаваться безотносительно к коррекции старого сопровождения. Переменной x9 присваивается список всех утверждений, относящихся к контексту вхождения терма t. Проверяется, что он содержит P. Переменной x24 присваивается результат обработки утверждений списка x22 нормализатором "нормодз"; в накопитель x23 при этом передаются использованные дополнительные утверждения. В список x7 заносится пара (x24, t'). Далее переменной x25 присваивается список всех утверждений списка x24, не вошедших ни в P, ни в x9. Чтобы обеспечить сопровождение для t', эти утверждения должны быть введены в ту точку задачи, где они являлись бы следствием ранее имевшихся утверждений. Для определения данной точки создается список x26 тех утверждений, следствием которых заведомо являются утверждения из x25. Он составлен из утверждений списка P, обеспечивающего о.д.з. для t; содержащихся в x9 утверждений списка x12 (т.е. всех реально использованных приемом для обоснования корректности замены t на t' утверждений), а также из утверждений накопителя x23, использованных при получении x24. Для списка x26 рассматриваются следующие подслучаи:

1. x26 включается в список посылок задачи. Тогда все утверждения из x25, не являющиеся посылками задачи, заносятся в список посылок. Затем просматриваются утверждения F списка P, входящие в список посылок, и те из них, которые далее не нужны для сопровождения по о.д.з., но являются следствиями прочих посылок, удаляются.

2. x26 включается в объединение списка посылок с условиями задачи на описание, отличными от преобразуемого условия. Тогда выполняются действия, аналогичные действиям предыдущего пункта, но утверждения заносятся в список условий, и удаляются не используемые для сопровождения по о.д.з. условия.

3. x26 пересекается с каким-либо из логических контекстов, внутри которых расположено вхождение t. Берется наименьший такой контекст K, и утверждения списка x25 добавляются к его списку дополнительных утверждений. Те из утверждений списка P, которые входят в K и далее не нужны для сопровождения по о.д.з., но являются следствиями остающихся утверждений, удаляются из K.

После выполнения указанных действий - откат к продолжению перечисления вхождений x15 надтермов заменяемого терма. По завершении перечисления - переход через "иначе 1" к фрагменту, содержащему контрольную точку "прием(3 3)". Здесь переменной x16 присваивается результат применения выполняемой приемом замены к текущему терму задачи.

Далее предпринимается попытка исключения из комментария (сопровождение x14) элементов, далее не нужных для сопровождения по о.д.з. Просматриваются пары (P, t) списка x14, где t - выражение; P - список утверждений, обеспечивающих сопровождение по о.д.з. для t. Если t входит в преобразуемый терм задачи и не входит в результат x16 его преобразования, то проверяется, имеются ли еще какие-либо вхождения t в задачу, для которых данная пара дает сопровождение по о.д.з. Особо учитывается комментарий (контекст ...), в котором хранятся утверждения, отложенные до момента редактирования ответа. Если t больше нигде не встречается, то (P, t) заносится в список x17. По окончании цикла элементы списка x17 исключаются из комментария (сопровождение ...), а элементы набора x7 - регистрируются в этом комментарии. Рассматриваются утверждения F списков P, относящихся к элементам набора x17, которые не встречаются в списках P оставшихся элементов комментария (сопровождение ...). В списках посылок и условий задачи веса данных утверждений уменьшаются до 0. Это позволяет решателю перейти к преобразованиям утверждений, которые до текущего момента были "заморожены" как обеспечивающие сопровождение.

Далее - переход через "ветвь 1" к фрагменту, содержащему контрольную точку "прием(35)". Здесь происходит учет имеющихся в x8 информационных элементов (замечание A) и (комментарийипосылок A). В обоих случаях A - комментарий, который нужно зарегистрировать, соответственно, в списке комментариев к задаче либо к ее списку посылок. Этот комментарий может содержать служебное слово "результат"(отличное от заголовка комментария), вместо которого подставляется заменяющий терм x5. Далее комментарий регистрируется в задаче.

По окончании ввода комментариев - переход через "ветвь 1" к фрагменту, содержащему контрольную точку "прием(37)". Здесь рассматривается список x12 утверждений контекста замены, использованных приемом для обоснования своих действий. По комментариям (выводимо ...) к посылкам списка x12 составляется список x18 исходных посылок задачи, косвенным образом использованных приемом. Если замена делается в исходной посылке, то она сама добавляется к списку x18. Затем утверждения списка x18 добавляются к комментарию (выводимо ...), относящемуся к текущей посылке (если замена делается в посылке) либо ко всей задаче.

Далее - переход через "ветвь 1" к фрагменту, содержащему контрольную точку "прием(38)". Здесь выполняется общая коррекция комментариев задачи, необходимая при замене. Некоторые из комментариев могли ссылаться на надтермы T заменяемого терма t и требовать выполнения в них синхронной замены его на заменяющий терм t'. Обычно такие комментарии имеют вид $(A\ T)$, где A - логический символ, используемый в качестве заголовка. Реализуется цикл просмотра комментариев x21 этого вида, у которых T есть надтерм заменяемого терма. С помощью справочника "коммент" уточняется необходимость коррекции (ненулевое значение означает отмену коррекции, так что по умолчанию она выполняется). Затем реализуется замена в терме T подтерма t на t'. Чтобы избежать рассогласований, результат замены подвергается некоторой стандартизации, аналогичной той, которая будет применяться к x16 при выполнении приемом своего основного преобразования.

По окончании коррекции комментариев - переход через "ветвь 1" к фрагменту, содержащему контрольную точку "прием(39)". Если замена происходит в условии задачи на описание, то просматриваются такие отличные от переменных подтермы s заменяемого терма, которые не входят в заменяющий. Справочник "пересмотр" определяет по ним посылки и условия задачи, перестающие быть сопровождающими после замены, и уменьшает их веса до 0. В этом же цикле находятся неизвестные, входящие в заменяемый терм, но отсутствующие в заменяемом. Так как исключение неизвестной в одном из условий задачи может привести в тому, что оставшиеся условия с этой неизвестной дадут искомый ответ, веса их понижаются до 1. Это активизирует приемы усмотрения ответа. По завершении цикла - откат к переходу через "ветвь 1".

Если список информационных элементов x6 имеет элемент (установка $A\ B$), то находится цель задачи $(A\ C_1\ \ldots\ C_n)$, которая преобразуется в цель $(A\ C_1\ \ldots\ C_n\ B)$.

Если список информационных элементов x6 имеет элемент (обозначения A), то для каждой переменной x списка A вводится новая цель задачи (обозначение x). Такие цели определяют переменные, вхождение которых в ответ недопустимо.

После очередного перехода через оператор "ветвь 1" появляется фрагмент программы, начинающийся с оператора "равно(x3 1)". Здесь процедура разветвляется: отдельно рассматриваются случаи замены в условии задачи и замены в посылке. Начнем с первого из них. Для задачи на описание, у которой заменяемый терм содержал неизвестные, а заменяющий не содержит их, веса всех прочих неизвестных условий уменьшаются до 1. Это несколько усиливает уже предпринимавшиеся выше аналогичные действия. Затем - переход через "ветвь 2".

Здесь предпринимается окончательная стандартизация терма x16, на который будет заменено текущее условие задачи. Она состоит в упорядочении операндов коммутативных операций, расположенных в заменяющем терме x5 либо на пути от корня заменяемого терма к максимальному включающему его логическому контексту (последний контекст списка x11). Результат стандартизации присваивается переменной x21.

Если имеется комментарий (внимание A) к посылкам задачи, то предпринимается его коррекция. Набор A состоит из троек $(B_1\ B_2\ B_3)$, определяющих такие наборы утверждений B_1, что при занесении любого из них в список посылок нужно уменьшить до величины B_3 вес терма задачи B_2. При совпадении B_2 с заменяемым условием происходит замена его в данной тройке на x21. Далее - переход через "ветвь 1" к фрагменту, в котором аналогичная коррекция выполняется для комментариев (внимание A) к текущей задаче на описание.

Если список x6 содержит элемент "стандследствие", то в комментарии (стандследствие ...) регистрируется информация о выводимости исходной версии текущего условия задачи из ее новой версии x21, пополненной списком x12 утверждений, использованных приемом. Обычно эти действия выполняются в ситуации, когда основное сопровождение по о.д.з. нарушено (например, при редактировании ответа), но информация для проверки условий на о.д.з. может еще понадобиться.

Если текущая задача имеет тип "преобразовать" и для нее был введен комментарий (коррекцияпосылок A), то рассматривается элемент (коррекцияпосылок B) из x6, и список A пополняется элементами списка B. Такие действия позволяют накапливать в комментарии (коррекцияпосылок A), созданным в задаче некоторым внешним пакетным нормализатором, всю информацию о сопровождении по о.д.з., которая может понадобиться впоследствии.

Наконец, приходим к фрагменту, содержащему контрольную точку "прием(42)". Здесь реализуется замена текущего условия задачи на терм x21. Предварительно переменной x22 присваивается вхождение в x21 заменяющего терма x5. Если x21 совпадает с текущим термом задачи, то замена не выполняется, и выход из оператора по истинностному значению "ложь".

Если x6 имеет элемент (примечание A), то все элементы набора A присоединяются к списку комментариев измененного условия. В случае редактирования ответа задачи на описание удаляются все комментарии (параметры …) к измененному условию, чтобы разблокировать повторное упрощение его известных подвыражений. Если x6 имеет элемент (выводусловия A B), то утверждение A заносится в список условий и сопровождается комментариями B. Если x6 имеет элемент (удалениеусловия A), причем утверждение A входит в список условий текущей задачи на описание, то оно удаляется из этого списка. Предварительно проверяется отсутствие в x6 элемента (норм C D), блокирующего удаление утверждения C. Блокировка имеет место, если C упоминается хотя бы в одном элементе (выводимо …) списка D. Если в x6 имеется элемент (вхождение A), то A заменяется на вхождение измененного условия во внешний список. Таким образом информация о новом вхождении текущего условия в список условий (или в задачу) передается внешним процедурам.

Если x6 содержит логический символ "норм", то в x21 находится непосредственный надтерм T заменяющего терма, переменной x24 присваивается его заголовок, и определяется результат R применения к T нормализатора общей стандартизации выражений с заголовком x24. Затем, снова при помощи оператора "заменавхождения", выполняется замена T на R, и в данной ситуации реализация процедуры "заменавхождения" завершается.

Если x6 содержит элемент "нормализатор", то прием выполняет общую стандартизацию. Два таких преобразования подряд, относящихся к одному и тому же условию (что распознается с помощью комментариев "нормализатор"), указывают на целесообразность обработки нормализаторами общей стандартизации всего условия сразу. Здесь используется процедура "блокнормализации".

На этом ветвь процедуры "заменавхождения", обеспечивающая замену в условии, завершается. Возвращаемся к точке разветвления и переходим к рассмотрению случая замены в посылке (пункт "Случай замены в посылке" - "Исходная точка"). Как и выше, переменной x21 присваивается окончательная версия терма, на который будет заменена текущая посылка.

Если текущая задача имеет тип "исследовать", то предпринимается коррекция ее комментариев (прообраз f A), (условие g A). Первый из них - комментарий к посылке f либо к некоторой посылке списка A, означающий, что утверждение f является следствием утверждений списка A. Второй - комментарий к некоторой посылке списка A, означающий, что условие g внешней задачи на описание является следствием посылок A текущей задачи. При коррекции отдельно рассматриваются случаи, когда изменяемая посылка равна f либо входит в A. Данные комментарии необходимы, чтобы можно было проследить по блоку анализа ход выводов "в обратном направлении" (т.е. там, где делались эквивалентные переходы). Такая обратная цепочка выводов может позволить после возвращения к внешней задаче на описание и занесения в нее найденных следствий (например, значений неизвестных) отбросить избыточные условия. После коррекции комментариев - откат к переходу через "ветвь 1".

Как и для случая замены в условии, предпринимается коррекция комментариев посылок (внимание ...). Если текущая задача на преобразование возникла при реализации внешнего нормализатора, то с помощью комментария (нормализация A) находится список A комментариев данного нормализатора. В A берется элемент (коррекцияпосылок ...), и в нем регистрируется информация о выводимости новой версии посылки из старой. Такая же коррекция предпринимается для комментария (коррекцияпосылок ...) текущей задачи на преобразование.

Наконец, в фрагменте с контрольной точкой "прием(27)" выполняется замена текущей посылки задачи на утверждение x21.

Если задача имеет комментарий к посылкам "смисточник", то решение ее выполняется в специальном режиме "с расчисткой". Решение повторяется дважды, причем на первом цикле находятся посылки, фактически использованные при получении ответа, а на втором происходит выдача на экран только шагов получения таких посылок. После оператора "входит(смисточник комментарииипосылок(x22))" предпринимается коррекция структур данных, необходимых для выявления фактически использованных посылок. Затем - переход через "ветвь 2".

Так как задача изменилась, буфер обращений к идентифицирующим операторам сбрасывается.

После контрольной точки "прием(58)" предпринимается учет комментариев посылок (пассив A_1 A_2 A_3 A_4). Такие комментарии создаются, чтобы переключить внимание после вычисления выражения A_1. A_3 - посылка задачи, вес которой должен быть в этом случае уменьшен до A_4. Если x21 имеет конъюнктивный член вида "равно(A_1 t)", то выполняется уменьшение веса данной посылки.

Остальные действия для случая замены в посылке аналогичны случаю замены в условии.

2.1.2 Процедура "вывод"

Процедура используется для вывода следствий в посылках задачи. По частоте применения она сопоставима с процедурой "заменавхождения", однако эта частота сильно зависит от предметной области. В "алгебраических" разделах преобладает процедура "заменавхождения", в "геометрических" - процедура "вывод". Формат обращения к ней имеет вид "вывод(x1 x2 x3)". Здесь x1 - задача; x2 - утверждение, присоединяемое к ее списку посылок; x3 - набор информационных элементов, дополняющих и уточняющих реализацию процедуры. Типы этих элементов - те же, что у процедуры "заменавхождения". Их список можно найти в информации по логическому символу "заменавхождения".

Для рассмотрения программы процедуры используем пункт "Общие процедуры, используемые в приемах" - "Процедура ВЫВОД" оглавления программ.

В качестве x2 может выступать логическая константа, представленная как логический символ. Тогда она сразу преобразуется в формат терма. Операнды коммутативных операций в x2 стандартным образом переупорядочиваются.

Если в x3 имеется информационный элемент (выводимо 0), сигнализирующий о невозможности обосновать какие-то используемые при выводе утверждения, то преобразование блокируется. Этот технический прием используется крайне редко. При наличии в x3 элемента "выход" заблокированная процедура выдает истинностное значение "истина", иначе - "ложь".

Проверяется отсутствие комментария (приемы ...) к посылкам исходной задачи, блокирующего применение данного приема вывода.

Проверяется невхождение утверждения x2 в список посылок. Если x2 - кванторная импликация, то проверяется отсутствие в списке посылок другой кванторной импликации, полученной из x2 переобозначением связанных переменных. Проверяется, что x2 не является логической константой "истина".

Выполняется обращение к процедурам "учетприменения", "символы", регистрирующим факт применения в данной задаче данного приема.

Если выводится утверждение x2, имеющее такие свободные переменные, которые не встречаются в списке посылок, то выполняется коррекция комментария (новаяпеременная A_1 A_2) к посылкам текущей задачи. Здесь A_1 - список всех вспомогательных переменных, появившихся при выводе следствий. A_2 - набор той же длины, что и A_1, у которого на позиции, соответствующей переменной x набора A_1, находится список $S(x)$ всех введенных до x вспомогательных переменных, от значений которых может зависеть значение x. Новые переменные x утверждения x2 добавляются в конец списка A_1, причем списки $S(x)$ для них составляются из всех вспомогательных переменных, упоминаемых в утверждениях элементов (выводимо ...) набора x3, а также из всех новых переменных утверждения x2.

Если в наборе x3 присутствует элемент (новыйсимвол x A), то текущая задача Z представляет собой блок анализа некоторой задачи на описание Z'. Новая переменная x, вводимая приемом, присоединяется к списку неизвестных задачи Z, а определяющие ее утверждения списка A добавляются к списку посылок задачи Z'. По такой схеме вводятся вспомогательные объекты (точки, прямые, и т.п.) при дополнительных построениях в геометрических задачах.

Если в наборе x3 присутствует элемент (вспомпараметр x A), то текущая задача Z представляет собой либо блок анализа некоторой задачи на описание Z', либо имеет тип "доказать". Прием вводит некоторый новый параметр x, который предполагается временно рассматривать как известный и выразить через него неизвестные. По достижении такой цели параметр будет переброшен в разряд неизвестных. Если Z - задача на исследование, то переменная x регистрируется в ее комментарии (вспомпараметр ...), перечисляющем "временно известные" параметры. При этом утверждения списка A, определяющие x, добавляются к списку посылок задачи Z'. Если же Z - задача на доказательство, то переменная x регистрируется в ее комментарии (известно ...). В обоих случаях веса всех утверждений задачи обнуливаются.

Если в наборе x3 присутствует элемент (вспомнеизвестная x), то текущая задача Z представляет собой либо блок анализа некоторой задачи на описание Z', либо имеет тип "доказать". Прием вводит новую переменную, которая будет временно рассматриваться как неизвестная, равноправная с ранее введенными неизвестными. Если Z - блок анализа, то переменная x присоединяется к спискам неизвестных как задачи Z, так и задачи Z'. Каждая из этих задач сопровождается комментарием (вспомнеизвестная x). Если Z - задача на доказательство, то переменная x регистрируется в ее комментарии (неизвестные ...). Веса посылок и условия задачи Z заменяются на 0. Смысл разбиения переменных задачи на "известные" и "неизвестные" состоит здесь в управлении процессом выражения одних ее переменных через другие.

Если в наборе x3 присутствует элемент (геомредактор A), то набор A содержит список инструкций по преобразованию чертежа, сопровождающего задачу. Эти инструкции передаются процедуре "допчертежа", перечисляющей различные варианты размещения новых элементов чертежа и оценки этих вариантов. Отбирается лучший вариант, регистируемый в комментарии (геомредактор ...) к посылкам текущей задачи. Данный комментарий хранит описание текущего чертежа.

Если исходная задача имеет комментарий посылок (трассперечисл A_1 A_2 A_3), то при срабатывании приема ГЕНОЛОГа, имеющего ссылку (A_1,A_2,A_3), должен быть установлен режим трассировки по срабатываниям приемов на уровне текущей задачи. При наличии в x3 такой ссылки реализуется переустановка режима трассировки.

Вводится индикатор x4 регистрации в списке посылок новых утверждений, и начинается просмотр конъюнктивных членов x5 утверждения x2 для регистрации их в списке посылок. Особо учитывается случай наличия в x3 информационного элемента (прообраз ...). Тогда конъюнкция x2 не разбивается на свои конъюнктивные члены, а заносится в список посылок целиком. Она будет разбита на составляющие части отдельным приемом, который преобразует и сопровождающую информацию о выводимости.

Собственно заносимое в посылки утверждение, расположенное по вхождению x5, присваивается переменной x6. Проверяется, что оно отлично от логической константы "истина". Если исходная задача имеет комментарий к посылкам (вычерк A_1 A_2), то A_1 - ссылка на прием ГЕНОЛОГа; A_2 - утверждение, вывод которого приемом A_1 блокируется. Если $A_2 = 0$, то любые срабатывания приема A_1 блокируются. Проверяется, не блокирует ли данный комментарий регистрацию утверждения x6.

Если имеется комментарий (вывод A) к посылкам текущей задачи, то в его наборе A перечисляются утверждения, ранее заносившиеся в список посылок процедурой "вывод". Повторное занесение утверждения x6 блокируется. Если x6 - кванторная импликация, то проверяется отсутствие в A кванторной импликации, отличающейся лишь переобозначением переменных. Затем x6 регистрируется в данном комментарии; указатель x4 устанавливается на 1, и x6 заносится в список посылок задачи x1. Если x6 - кванторная импликация, то она снабжается комментарием "новый", блокирующим применение ряда специальных приемов.

При наличии в x3 элементов (удалениепосылки A) предпринимается исключение тех содержащихся в списках A посылок, которые не используются как сопровождающие по о.д.з.

Для отладочных целей используется комментарий (текпосылка A). У него A - накопитель, сохраняющий пары (утверждение - ссылка на прием ГЕНОЛОГа, создавший это утверждение) для пятнадцати самых длинных посылок, выведенных при решении задачи. Предпринимается рассмотрение x6 для коррекции данного комментария.

Корректируются комментарии (смисточник ...), (повтор ...), необходимые для упоминавшегося выше режима решения задачи "с расчисткой". Затем - откат к переходу через "ветвь 1".

Возникает фрагмент, начинающийся с оператора "смпосылка(...)", который корректирует структуры данных, используемые идентифицирующими операторами для быстрого поиска в списке посылок.

Как и в случае оператора "заменавхождения", при регистрации равенства в задаче на исследование, имеющей цель "известно", анализируется наличие комментария (пассив ...). При наличии комментария посылок (новаяпосылка ...) проверяется необходимость понижения веса указанной в нем посылки. Сбрасывается буфер результатов обращения к идентифицирующим операторам.

Если x3 имеет элементы (примечание ...), то новая посылка снабжается указанными в них комментариями. Переменной x7 присваивается список исходных посылок задачи, использованных при выводе, и новая посылка сопровождается комментарием (выводимо x7). Вводится также комментарий "следствие", позволяющий отличить новые посылки от исходных (быть может, преобразованных).

Если текущая задача не имеет типа "исследовать" и не является задачей на доказательство, относящейся к геометрии, то веса всех посылок, имеющих с x6 общую свободную переменную, понижаются до 1. В случае задачи на преобразование, условие которой имеет с x6 общую свободную переменную, таким же образом понижается вес условия.

Если x3 имеет информационный элемент (прообраз A B), где B отлично от 0, то посылка A, использованная при выводе, является следствием утверждений списка B, остальных использованных при выводе утверждений и нового утверждения x2. Информация об этом регистрируется в комментариях (прообраз ...).

По окончании цикла обработки конъюнктивных членов x6 утверждения x2 находятся элементы (вывод ...), перечисленные в набор x3, и реализуется определяемый ими вывод дополнительных утверждений. Здесь используется рекурсивное обращение к процедуре "вывод". Далее, если индикатор x4 равен 1 (т.е. возникли новые посылки), происходит выход из процедуры по истинностному значению "истина", иначе - выход по значению "ложь".

2.1.3 Процедура "выводусловия"

Процедура "выводусловия" применяется весьма редко. Это объясняется тем, что условия задачи на описание невыгодно перегружать избыточной информацией. Обычно они либо преобразуются эквивалентным образом, либо к ним добавляются выражающие ответ утверждения, полученные путем вывода следствий в блоке анализа. Лишь в особых случаях, когда непосредственные эквивалентные преобразования, разрешающие условия, невозможны, а добавляемое условие открывает реальный путь к последующему их разрешению, реализуется вывод дополнительных условий задачи. Кроме того, добавляемое условие часто представляет собой дизъюнкцию для разбора случаев.

Формат обращения к процедуре имеет вид "выводусловия(x1 x2 x3)". Здесь x1 - задача; x2 - утверждение, присоединяемое к ее списку условий; x3 - набор информационных элементов, дополняющих и уточняющих реализацию процедуры. Типы элементов - те же, что у процедуры "заменавхождения". Так как процедура используется редко, программа ее не выполняет такого объема дополнительных действий, как программы "заменавхождения" и "вывод". Эту программу находим в том же разделе оглавления программ, что и предыдущие.

Проверяется отсутствие комментария (выводусловия x2), указывающего, что утверждение x2 уже заносилось ранее в список условий. Такой комментарий вводится. Проверяется, что x2 отсутствует в списке условий и отлично от константы "истина". Затем оно заносится в список условий. Если в x3 имеется элемент (примечание A), то все элементы набора A становятся комментариями к новому условию. Переменной x4 присваивается список исходных посылок, использованных при выводе условия. Этот список регистрируется в комментарии (выводимо ...) к текущей задаче. Данный комментарий аккумулирует все исходные посылки задачи, использовавшиеся при преобразовании ее условий.

Если в x3 имеется элемент (задача Z), то это означает, что прием, перед тем, как создать выводимое утверждение, решал вспомогательную задачу Z. Если тип задачи Z - "преобразовать", то предпринимается попытка усмотреть среди ее посылок такие утверждения, которые могли бы оказаться полезными в текущей задаче как сопровождающие, по о.д.з. либо в более широком смысле. Для этого просматриваются все ее посылки x8, не входящие ни в список посылок, ни в список условий текущей задачи. Отбираются те из них, которые имеют хотя бы одну свободную переменную, причем каждое входящее в них выражение входит также в утверждение x2. По комментариям (выводимо ...) определяется список S всех исходных посылок задачи Z, следствием которых является x8. Если S содержится в списке посылок текущей задачи, то x8 заносится в ее посылки; если он содержится в объединении списков посылок и условий, то x8 заносится в список условий.

С помощью процедур "учетприменения", "символы" выполняется регистрация факта применения данного приема в данной задаче.

Если в x3 имеется элемент (вывод ...), определяющий дополнительно выводимые утверждения, то они заносятся в список посылок (если все использованные приемом утверждения суть посылки) либо в список условий. Далее - выход из процедуры.

2.1.4 Процедура "заменавхождений"

Если при выполнении приема замены есть основания предполагать, что заменяемый терм t имеет много вхождений в задачу, причем все они должны быть заменены на заменяющий терм s, то используется процедура "заменавхождений". Она находит все вхождения в задачу терма t, контекст которых содержит обоснование допустимости замены их на s, и выполняет такую замену. Формат обращения к процедуре имеет вид "заменавхождений(x1 x2 x3 x4)", где x1 - задача; x2 - вхождение первого символа заменяемого терма t (неважно в какой терм); x3 - заменяющий терм s. x4 - набор информационных элементов, дополняющих и уточняющих реализацию процедуры. Типы элементов - те же, что у процедуры "заменавхождения".

Программу процедуры можно найти в том же разделе оглавления программ, где находятся предыдущие процедуры. Она совсем небольшая. Проверяется различие термов s, t; находится список x5 утверждений, составляющих обоснование замены; проверяется, что s не является подтермом терма t, и просматриваются все вхождения в задачу терма t. Если их контекст содержит x5, предпринимается обращение к процедуре "заменавхождения", выполняющей замену для данного вхождения. Переменная x7 играет роль индикатора замены. Если хотя бы одна замена имела место, то x7 становится равна 1. В этом случае выход из процедуры происходит по истинностному значению "истина"; в противном случае - по значению "ложь".

2.1.5 Процедура "заменагруппы"

Если прием должен заменить сразу группу посылок либо группу условий задачи на одно новое утверждение, то используется процедура "заменагруппы". Формат обращения к ней имеет вид "заменагруппы(x1 x2 x3 x4 x5)", где x1 - набор заменяемых посылок при x2 $=$ 0 либо набор заменяемых условий при x2 $=$ 1. x3 - преобразуемая задача; x4 - заменяющее утверждение. x5 - набор информационных элементов, дополняющих и уточняющих реализацию процедуры. Типы элементов - те же, что у процедуры "заменавхождения".

Программа процедуры находится через оглавление программ. Она начинается с регистрации факта применения приема в данной задаче (процедуры "учетприменения" и "символы"). Проверяется отсутствие комментария (приемы ...), блокирующего применение данного приема. Заменяющее утверждение x4, если оно было представлено в формате логического символа (т.е. константы "истина" либо "ложь"), преобразуется в формат терма. В списке x1 заменяемые утверждения заменяются на их вхождения, соответственно, в список посылок либо в список условий. Переменной x8 присваивается список всех исходных посылок, использованных приемом. В случае задачи на исследование исключаются все комментарии (прообраз ...) к остающимся посылкам, в которых есть ссылка на заменяемые посылки.

Если x5 содержит элемент (новыйсимвол ...), указывающий новые объекты, вводимые приемом, то он обрабатывается так же, как в процедуре "вывод".

Если x5 содержит элемент (посылка A_1 A_2), то проверяется, что все утверждения, использованные приемом, суть посылки. В этом случае утверждение A_1 заносится в список посылок и сопровождается комментариями A_2. Далее раздельно рассматриваются случаи замены посылок и условий.

Если заменяются посылки, то прежде всего добавляется новая посылка x4. Корректируются структуры данных, используемые идентифицирующими операторами. Новая посылка сопровождается комментариями, извлекаемыми из имеющихся в x5 элементов (примечание ...). Для нее также создается комментарий (выводимо x8). Далее происходит удаление всех заменяемых посылок. На основе элементов (вывод ...) из набора x5 осуществляется вывод дополнительных утверждений.

Если заменяются условия, то действия аналогичны: сначала водится новое условие, а затем отбрасываются заменяемые. Реализуется учет элементов (удалениеусловия ...) и (вывод ...), имеющихся в списке x5.

2.1.6 Процедура "попытказамены"

Если есть опасения, что предлагаемая приемом замена может завести задачу в тупик, то реализуется "осторожный" режим. Вместо того, чтобы сразу изменить текущую задачу Z, создается ее копия Z', и замена делается в копии. После этого предпринимается попытка довести до ответа задачу Z'. Если ответ получен, то он передается в задачу Z. Иначе - решение задачи Z продолжается без применения данного приема. Указанные действия выполняются процедурой "попытказамены(x1 x2 x3 x4 x5)". Смысл входных данных тот же, что в процедуре "заменавхождения(x1 x2 1 x3 x4 x5)". Иными словами, (x1, x2, 1) - координата вхождения заменяемого подтерма условия задачи x3; x4 - заменяющий терм; x5 - набор информационных элементов, уточняющих преобразование.

Входим в программу процедуры через оглавление программ и начинаем прослеживать выполняемые действия. Прежде всего, проверяется, что тип задачи - "доказать" либо "описать". Чтобы заблокировать повторение неудачной попытки замены, используется "защелка", реализованная через комментарий (попытказамены ...). Переменной x6 присваивается копия текущей задачи. Если в наборе x5 имеется элемент (Замечание A), то список A присоединяется к комментариям задачи x6. Переменным x8 и x7 присваиваются, соответственно, заменяемое вхождение в задаче x6 и вхождение того терма задачи x6, внутри которого оно расположено. Вводится комментарий (заменавхождения x8 x7 x4 x5) задачи x6, и предпринимается обращение к решению этой задачи. Максимальный уровень обращения - такой же, как у текущей задачи Z. Первый же шаг решения задачи x6 будет состоять в извлечении комментария (заменавхождения x8 x7 x4 x5) и обращении к процедуре "заменавхождения(x8 x7 1 x6 x4 x5)".

Если на задачу x6 получен ответ x9, отличный от символа "отказ", то определяется список использованных при ее решении исходных посылок, который передается во внешнюю задачу x3. Если x3 - задача на доказательство, то ее условие заменяется на "истина", иначе - ее список условий заменяется на множество конъюнктивных членов ответа x9. Задача x3 сопровождается комментарием "ответ", инициирующим немедленную выдачу ответа, и выход из процедуры по значению "истина". Если на задачу x6 получен "отказ", то - выход по значению "ложь".

2.1.7 Процедуры "попыткаспуска", "Попыткаспуска"

Если заменяется условие задачи на доказательство или описание, либо группа условий задачи на описаний, причем требуется "осторожный" режим, то используются, соответственно, процедуры "попыткаспуска" и "Попыткаспуска".

Обращение к первой из них имеет вид "попыткаспуска(x1 x2 x3 x4)". Здесь x1 - задача; x2 - вхождение ее условия в список условий либо в задачу; x3 - новое условие; x4 - набор информационных элементов, уточняющих преобразование. Обращение ко второй процедуре имеет вид "Попыткаспуска(x1 x2 x3 x4)"; единственное отличие состоит в том, что x2 - не единственное условие, а набор условий. Программы обеих процедур аналогичны рассмотренной выше программе "попытказамены".

2.1.8 Процедура "обозначение"

В задачах на описание, имеющих цель "программа", требуется определить систему утверждений, определяющих последовательное вычисление значений неизвестных. Иногда здесь бывает полезно выделить некоторое подвыражение t (вообще говоря, содержащее неизвестные), выбрать для него в качестве обозначения новую переменную x, и занести в список условий дополнительное равенство $x = t$, объявив переменную x неизвестной. При этом все вхождения t в условия задачи заменяются на x, а посылка $x = t$ сопровождается комментарием "блок", блокирующим обратную замену. Задача сопровождается комментарием (вспомпараметр x). Все эти действия реализуются процедурой "обозначение(x1 x2)", где x1 - текущая задача на описание; x2 - выражение t.

2.1.9 Процедура "учетприменения"

Процедура используется для учета срабатывания приема в комментариях к посылкам исходной задачи. К ней обращаются различные операторы, реализующие основное действие приема (замена подтерма, вывод следствия, и т.п.). Сохраненная в комментариях информация будет использована соответствующими общими процедурами, запускаемыми по окончании решения задачи. Формат обращения к процедуре имеет вид "учетприменения(x1 x2 x3 x4)". Здесь x1 - исходная задача; (x2, x3, x4) - ссылка на прием ГЕНОЛОГа: x2 - логический символ; x3 - номер узла статьи символа x2; x4 - заголовок приема.

На период выполнения процедуры счетчик шагов интерпретатора ЛОСа отключается. Процедура рассматривает следующие комментарии к посылкам исходной задачи:

1. Комментарий (список A), перечисляющий в наборе A установки (B_1 B_2 B_3 B_4) на регистрацию срабатываний приема в буфере задачника. Здесь (B_1 B_2 B_3) - ссылка на прием ГЕНОЛОГа; B_4 - индикатор срабатывания. Если указанная ссылка совпадает с (x2, x3, x4), то индикатор B_4 устанавливается на 1;

2. Комментарий (задача A_1 A_2), используемый в качестве накопителя информации для архива задачника. A_1 - накопитель логических символов, возникающих при решении в посылках и условиях задачи; A_2 - накопитель ссылок на сработавшие приемы, образованный парами (B_1 B_2). Здесь B_1 - заголовок приема; B_2 - набор пар (логический символ - набор номеров узлов приемов). Проверяется, что сработавший прием не является приемом нормализатора общей стандартизации, после чего ссылка на него регистрируется в A_2. Набор A_1 корректируется другой процедурой - "символы(...)". Архив задачника позволит найти все задачи, где срабатывал заданный прием, а также все задачи, при решении которых появлялся заданный логический символ.

3. Комментарий (учетприменения A), используемый процедурой вывода теорем для накопления информации о теоремах сработавших приемов. A - набор пар (логический символ - набор номеров теорем сработавших приемов, закрепленных за данным символом);

4. Комментарий (выписка A), используемый для накопления статистики о "холостом ходе" приемов и числе их срабатываний. A - набор пар (B_1 B_2), где B_1 - логический символ; B_2 - накопитель данных, относящихся к приемам ГЕНОЛОГа, закрепленным за этим символом. Этот накопитель состоит из пар (C_1 C_2), где C_1 - заголовок приема; C_2 - накопитель данных, относящихся к приемам символа B_1, имеющим заголовок C_1. Он состоит из троек (D_1 D_2 D_3), где D_1 - номер узла приема; D_2 - счетчик холостого хода, равный числу тысяч шагов работы интерпретатора, затраченных на попытки применения приема, не приведшие к его срабатыванию. Данная величина представлена в формате символьного числа. Счетчик включается при прохождении через оператор "контрольприема(...)" и корректируется при откате. Это делается интерпретатором ЛОСа автоматически; нужно лишь создать комментарий (выписка ...) к посылкам исходной задачи и использовать оператор "трассировка(контроль 1)" для включения режима контроля холостого хода. Значение D_3 - счетчик числа применений приема. Именно оно и корректируется процедурой "учетприменения".

2.2 Завершение решения задачи

Перечислим основные процедуры, используемые при завершении решения задачи, обрыве решения либо откате.

2.2.1 Процедура "редакторответа"

После того, как прием усматривает, что задача на описание решена, он обычно обращается к процедуре "редакторответа", выполняющей завершающую обработку условий задачи перед выдачей ответа. Такое обращение необходимо во всех случаях, когда комментарии задачи могут содержать фрагменты ответа, относящиеся к ранее исключенным неизвестным. Кроме того, оно необходимо для учета целей, определяющих упрощение ответа (например, для явного разрешения условий на известные параметры).

Формат обращения к процедуре имеет вид "редакторответа(x1 x2 x3)". x1 - текущая задача на описание, x2 - набор дополнительных ограничений на известные переменные, вытекающих из условий с неизвестными. Переменной x3 присваивается окончательный ответ задачи x1. Непустое значение x2 встречается редко; оно определяется теоремами приемов "ответ(...)" по стандартному виду явного описания. Например, если условия задачи содержат описание $a \in x, x \subseteq b$ неизвестного множества x, то в список x2 включается утверждение $a \in b$.

Выходим на начало программы "редакторответа" через тот же раздел оглавления программ, в котором размещены процедуры преобразования задач.

Прежде всего, контролируются цели (обозначение x), указывающие переменные x, которые не должны входить в ответ. Если некоторая такая переменная встречается в списке условий, процедура выдает "отказ" (в качестве значения своей выходной переменной x3).

Если задача имеет цель "исследовать", то ее решение состояло в накоплении списка утверждений, дающих требуемую общую характеризацию неизвестных. Так как новые утверждения заносятся в начало списка условий, то для сохранения хронологической последовательности утверждений их порядок меняется на обратный, после чего выдается их конъюнкция.

В остальных случаях переменной x4 присваивается список условий задачи, пополненный утверждениями набора x2. Проверяется необходимость обращения к вспомогательной задаче на описание для завершающего редактирования ответа. Основными признаками, указывающими на необходимость применения вспомогательной задачи, служат наличие комментария (контекст ...), либо комментария (сокращнеизв ...), либо наличие цели "упростить".

Если вспомогательную задачу вводить не нужно, то переход через "иначе 1". Здесь анализируется случай, когда задача имеет цель "пример". Если она имеет также цель "полный", то проверяется, что все условия задачи, не содержащие неизвестных, являются следствиями ее посылок. При неудаче выдается "отказ". Иначе комментарий (выводимо ...) пополняется посылками, использованными при проверке, и в качестве ответа выдается конъюнкция всех условий, содержащих неизвестные.

Если задача с целью "пример" не имеет цели "полный", то переход через "иначе 2". Здесь реализуется цикл просмотра условий задачи, не содержащих неизвестных. Предпринимается попытка усмотреть, что такое условие является следствием остальных условий и посылок. При успехе условие удаляется из списка условий и из заготовки ответа x4. По завершении цикла - откат к переходу через "ветвь 1".

Если задача не имела цели "пример", а также по завершении указанного выше цикла попыток отбросить известные условия, - выдается ответ. Он представляет собой конъюнкцию всех не входящих в посылки утверждений списка x4.

Если было принято решение о создании вспомогательной задачи для редактирования ответа, то прежде всего анализируется комментарий (сокращнеизв A_1 A_2). Он возник при решении внешней задачи A_1 с несколькими неизвестными, когда временно была сохранена лишь одна неизвестная, а остальные объявлены известными. A_2 - список таких "временно известных" переменных. Наличие данного комментария означает, что реализован лишь первый этап решения, и речь идет не о редактировании ответа, а о продолжении решения после того, как одна из неизвестных была выражена через прочие неизвестные.

После контрольной точки "прием(2)" переменной x6 присваивается A_2, переменной x7 - набор всех содержащих неизвестные текущей задачи утверждений списка x4; переменной x8 - набор остальных утверждений списка x4. Составляется список x9 таких утверждений из x7, что выполнение их может повлечь какие-то невырожденные дополнительные ограничения на значения переменных списка x6. К набору x8 добавляется утверждение существования таких значений неизвестных текущей задачи, при которых истинна конъюнкция элементов списка x9. Теперь x8 будет играть роль списка условий, которые нужно разрешить относительно "оставшихся неизвестных" списка x6, в то время как x7 является отложенным до момента редактирования фрагментом ответа. С помощью справочника "сокращнеизв" создается список целей вспомогательной задачи на описание x10, имеющей список условий x8 и прежний список посылок. Те несущественные неизвестные задачи x10, которые используются в x7, рассматриваются при решении задачи x10 как существенные. Фрагмент ответа x7 регистрируется в комментарии (контекст ...). Затем предпринимается попытка решить задачу x10. Если на нее получен "отказ", то выдается "отказ" на текущую задачу. Иначе - предпринимается перенесение из x10 в текущую задачу тех комментариев, которые могут пригодиться впоследствии. Если найденный ответ R содержит переменные x, упомянутые в комментариях (вспомпараметр x), то выдается отказ. В противном случае выдается ответ R.

Если комментария (сокращнеизв ...) не было, то переход через "ветвь 2". Здесь будет происходить собственно редактирование ответа. Рассматриваются два подслучая.

Первый из них - наличие цели (серия $n_1 \ldots n_k$). Тогда данная задача возникла при редактировании некоторого внешнего параметрического описания, причем $n_1, \ldots, n_k$ - все его параметры. Чтобы завершить редактирование, нужно разрешить относительно $n_1, \ldots, n_k$ известные утверждения списка x4. Иначе условия на параметры сохранят неявный вид.

Переменной x7 присваивается список тех параметров $n_1, \ldots, n_k$, которые явным образом встречаются в содержащих неизвестные утверждениях списка x4, а также в фрагментах ответа из комментариев (контекст ...). Условия на эти параметры должны явным образом присутствовать в параметрическом описании. Переменной x8 присваивается список остальных параметров; они могут в параметрическом описании быть отброшены. Вводится вспомогательная задача x9, условиями которой служат все не содержащие неизвестных утверждения списка x4. Неизвестными задачи x9 являются все параметры $n_1, \ldots, n_k$, причем переменные списка x8 объявлены несущественными неизвестными. Ей передаются комментарии (контекст ...), в которых регистрируются также все содержащие неизвестные утверждения списка x4. В зависимости от наличия цели "упростить", решение задачи x9 предпринимается до максимального уровня 9 либо 5. Ее ответ x10 выдается как результат работы процедуры. Если x10 отлично от символа "отказ", то предварительно в текущую задачу из задачи x9 передаются комментарии, которые могут понадобиться в дальнейшем.

Второй подслучай - общий случай редактирования ответа (в частности, вспомогательная задача x9 из предыдущего подслучая будет, в конце своего решения, обрабатываться через данный подслучай). Здесь вводится вспомогательная задача на описание x5, условиями которой являются все утверждения списка x4. Цели ее берутся из текущей задачи (кроме цели "перечисление"), и к ним добавляется цель "редакция". При отсутствии условий с неизвестными (кроме некоторых особых целевых установок) задаче x5 придается цель "проверка", инициирующая попытки довести ее ответ до логической константы. Если имеется комментарий (контекст ...), то сохраненные в нем фрагменты ответа добавляются к условиям задачи x5. В этом случае цели задачи x5 берутся из той задачи, которая была указана в комментарии, причем к ним добавляется символ "редакция". В зависимости от наличия цели "упростить", максимальный уровень задачи x5 берется равным 9 либо 5. По окончании решения выполняется перенесение в текущую задачу тех комментариев задачи x5, которые могут понадобиться в дальнейшем. Затем ответ задачи x5 выдается как результат работы процедуры.

2.2.2 Процедура "замещениеусловий"

Если при решении задачи на исследование Z, являющейся блоком анализа внешней задачи на описание Z', получены утверждения R, составляющие часть ответа последней задачи, либо выражающие одни неизвестные через другие, либо определяющие разбор случаев, то они переносятся в список условий задачи Z'. При этом предпринимается попытка исключить все условия U задачи Z', оказывающиеся следствиями утверждений R и остальных условий. Такое исключение избавляет от необходимости специально выполнять проверку условий U. Перечисленные действия (включая поиск условий U) выполняются процедурой "замещениеусловий(x1 x2)", используемой приемами перед обрывом рассмотрения блока анализа. Ее входные данные - текущая задача на исследование x1 и набор x2 посылок задачи x1, которые требуется перенести в задачу Z' (т.е. набор R). Чтобы определить список U, используются комментарии об обратной выводимости, сохраняемые в блоке анализа.

Процедура начинает свою работу с того, что обращается к оператору "блокзамещения", анализирующему обратные выводы в блоке анализа. Формат обращения к этому оператору имеет вид "блокзамещения(x1 x2 x3 x4)". Здесь x1, x2 - входные данные процедуры "замещениеусловий". Переменной x3 присваивается набор U условий задачи Z', являющихся следствиями утверждений списка x2, пополненных посылками задачи Z' и некоторыми посылками D задачи Z. Последние будут играть роль утверждений, сопровождающих утверждения списка x2 при занесении в условия задачи на описание. Переменной x4 присваивается тройка $(C_1\ C_2\ C_3)$, где C_1 - объединение D с посылками задачи Z', использованными для вывода U из x2; C_2 - набор некоторых посылок задачи Z', для которых по комментариям (прообраз ...) усматривается их выводимость из x2, D и посылок задачи Z'. C_3 - набор, длина которого равна длине набора C_2. На позиции этого набора, соответствующей утверждению F набора C_2, располагается список фактически использованных при выводе F из x2 утверждений D и посылок задачи Z'.

Рассмотрим программу оператора "блокзамещения". Переменной x5 присваивается набор наборов исходных посылок блока анализа, использованных при выводе посылок x2. Переменной x6 присваивается список всех условий внешней задачи на описание Z, переносившихся в блок анализа. Вводится накопитель x7 набора C_2, которому присваивается список всех утверждений списка x2, выведенных с использованием условий задачи Z' либо представляющих собой дизъюнкции, содержащие неизвестные задачи Z. Вводится накопитель x8 набора C_1, которому присваивается список всех исходных посылок блока анализа, использованных при выводе утверждений x2 и не являющихся условиями задачи Z'. Вводится накопитель x9 набора C_3. Его длина равна длине набора x7, а заполнен он пустыми словами. Вводится накопитель x10 набора U, инициализируемый пустым словом. Затем начинается цикл пополнения накопителей, состоящий в последовательной обработке элементов списка C_2. При этом переменная x12 хранит текущую обрабатываемую части списка C_2.

Сначала x12 равно x7; на очередном шаге цикла берется первый элемент x13 списка x12, исключаемый из списка. При рассмотрении данного элемента возможно занесение в начало списка x12 и одновременно в список x7 новых элементов.

Для текущего значения x13 - посылки блока анализа - просматриваются все его комментарии x15 вида (условие g A) либо (прообраз g A), у которых список A содержит x13. Каждый из этих комментариев означает выводимость утверждения g из A, причем в первом случае g является условием внешней задачи на описание, а во втором - посылкой блока анализа. Здесь речь идет об "обратной" выводимости, т.е. при выводе следствий в блоке анализа утверждение x13 возникло как следствие g и остальных утверждений списка A, причем сразу же было замечено, что и обратный вывод имеет место. В случае комментария (прообраз ...) проверяется, что g еще не занесено в список x7. Проверяется, что все утверждения списка A, отличные от константы "истина", суть посылки блока анализа.

Переменной x16 присваивается та часть списка A, которая не содержится в x7. Находятся подсписок x17 всех утверждений списка x16, которые были выведены при использовании условий внешней задачи на описание Z', а также список x18 всех использованных при получении x16 исходных посылок блока анализа, не являющихся условиями Z'. Если x17 непусто, то проверяется, что список x2 переносимых в Z' утверждений содержит равенство, явно определяющее значение какой-либо неизвестной, а все утверждения в x17 суть либо неравенства, либо указатели "число(...)" типа значения переменной. При этом элементы списка x17 добавляются к списку x18. Чтобы получить набор x20, который будет занесен на позицию списка C_3, соответствующую утверждению g, к x18 добавляются все утверждения, использованные, согласно уже заполненной части набора C_3, при выводе общей части списков A, x7.

Если x15 имеет вид (условие ...), причем утверждение g входит в текущий список условий задачи Z' и еще не зарегистрировано в x10, то оно добавляется к x10, а утверждения набора x20 добавляются к списку x8. Если x15 имеет вид (прообраз ...), то утверждение g заносится в списки x7 и x12, причем к концу списка C_3 добавляется x20.

По завершении цикла процедура "блокзамещения" выдает результаты $U =$ x10; $(C_1,C_2,C_3) =$ (x8,x7,x9).

Возвращаемся к программе процедуры "замещениеусловий". После обращения к процедуре "блокзамещения" определились значение переменной x3, равное списку U исключаемых условий задачи Z', а также значение переменной x4, равное тройке (C_1, C_2, C_3), определенной выше. Переменной x5 присваивается список C_1; переменной x6 - пересечение x2 и C_2; переменной x7 - список перенесенных в блок анализа условий задачи Z'. Далее x6 - накопитель фактически переносимых в список условий внешней задачи утверждений, из которого исключены утверждения, являющиеся следствиями ее посылок. x5 - накопитель списка посылок внешней задачи, использованных для обоснования выполняемой замены условий.

В качестве типичной ситуации, требующей обращения к процедуре, можно рассмотреть решение системы алгебраических уравнений. Тогда x2 - либо равенство, выражающее одну неизвестную через другие, либо дизъюнкция таких равенств. U - список всех условий внешней задачи, которые оказываются следствиями x2 и простых дополнительных посылок блока анализа (неравенств для сопровождения по о.д.з. и т.п.). Эти дополнительные посылки, вместе с использованными посылками внешней задачи, составляют список C_1. В списке C_2 перечислены всевозможные посылки блока анализа, выводимые из x2 с теми же дополнительными обоснованиями. Так как решается система уравнений, то обычно x2 является линейной комбинацией нескольких исходных уравнений, и каждое в отдельности такое уравнение не выводится из x2. Поэтому U сначала будет пустым. Чтобы получить непустой список U, нужно добавить к x2 несколько других уравнений (исходных условий или их следствий), имеющихся в блоке анализа. Желательно сделать это так, чтобы после регистрации расширенного x2 и удаления U получить возможно более простой список условий внешней задачи. Поэтому, после оператора "повторение", начинается цикл попыток пополнения списка x2. В этом цикле продолжается рассмотрение "обратного" вывода, оборвавшегося на утверждениях списка C_2. В отличие от цикла процедуры "блокзамещения", теперь разрешается использовать в качестве дополнительных посылок более широкий класс утверждений (например, уравнения). Переменной x13 присваивается список дополнительных посылок для очередного перехода по комментариям (условие ...), (прообраз ...). К объединению x14 списков x2 и x13 применяется процедура "блокзамещения", дающая новую версию значений U, C_1, C_2, C_3. Если оказывается, что эта версия приводит к более компактному результату преобразований списка условий внешней задачи, то x2, x4, x5, x6 заменяются на значения данной версии.

По завершении цикла пополнения списка x2 - переход через "иначе 1". Все утверждения набора x5, выведенные с использованием хотя бы одного условия внешней задачи, переносятся из x5 в x6.

Переходим через "ветвь 1"; здесь переменной x9 присваивается внешняя задача на описание. Если все неизвестные задачи x9 - числовые, то составляется список x10 всех ее неизвестных, которые либо явно выражены сотношениями списка x6, либо могут быть явно выражены путем решения линейных уравнений. Если после этого остаются какие-то неизвестные, не вошедшие в x10, то предпринимается попытка усмотреть в блоке анализа линейные уравнения, дающие их явное выражение через x10. Такие уравнения присоединяются к x6. Хотя список отбрасываемых условий это и не сокращает, но ход решения может сильно упростить.

Выполняется анализ сопровождения по о.д.з.; при необходимости список x6 и список посылок внешней задачи пополняются.

Наконец, реализуется замена условий во внешней задаче. Утверждения списка U (переменная x3) отбрасываются; утверждения списка x6 заносятся в список условий. Новые дизъюнкции помечаются комментарием "разборслучаев"; из блока анализа переносятся некоторые комментарии, сопровождавшие утверждения списка x6. На этом действия процедуры завершаются.

2.2.3 Процедура "свертка"

Данная процедура обеспечивает сокращенную переформулировку выражений. Она используется при завершающем редактировании ответа задачи на преобразование, не имеющей целей, блокирующих попытку сжатия текста ответа. Формат обращения к процедуре имеет вид "свертка(x1 x2 x3)", где x1 - преобразуемый терм; x2 - список посылок, относительно которых выполняются преобразования; x3 - набор комментариев. При обращении в x3 попадают необходимые для учета сопровождения по о.д.з. элементы (выводимо ...), (коррекцияпосылок ...). Впоследствии процедура добавляет к ним комментарий (задача Z'), создавая для технических целей фиктивную задачу Z' типа "доказать", имеющую список посылок x2 и условие x1. Она нужна только для обращений к справочнику "одз".

Если выражение x1 односимвольное, то оно выдается в качестве результата. Иначе x1 представляется в виде $f(t_1 \dots t_n)$, и переменной x5 присваивается набор $T_1, \dots T_n$

результатов применения данной процедуры к операндам $t_1, \ldots, t_n$. Определяется терм x7 вида $f(T_1 \ldots T_n)$. В список x2 и в комментарии набора x3 заносятся утверждения, необходимые для сопровождения по о.д.з. терма x7.

Собственно "сжатие" терма осуществляется специальными пакетными нормализаторами. Обычно их названия начинаются с приставки "упрощ": "упрощплюс", "упрощумножение", и т.п. Например, нормализатор "упрощумножение" может сгруппировать два сомножителя a^c, b^c, преобразовав их в сомножитель $(ab)^c$. Нормализатор общей стандартизации "нормстепень" делал бы в этой ситуации обратное преобразование. Чтобы найти нужный пакетный нормализатор, применяется справочник "нормупростить". По заголовку f он определяет название нормализатора N. Если $N \neq 0$, то выполняется обращение к нормализатору на входных данных x7, x2, x3, и выдается результат данного обращения. Иначе - выдается терм x7.

2.2.4 Процедуры "установка"и "Контроль"

Иногда бывает трудно оценить целесообразность применения приема в момент его срабатывания. Если есть основания предполагать, что это срабатывание, если вообще окажется полезным, приведет к некоторой ситуации, которую легко будет усмотреть через определенный промежуток времени, то можно применить режим отложенной фильтрации. Условие, определяющее желаемую ситуацию ("отложенный фильтр") будет проверяться здесь не сразу при срабатывании, а через указанное число шагов. Если это условие окажется нарушено, произойдет откат - будет восстановлено то состояние задачи, которое имелось перед применением приема.

Заметим, что отложенная фильтрация в решателе используется крайне редко - скорее это некоторый резерв, которым можно воспользоваться в особо сложных случаях.

Для реализации режима отложенной фильтрации используется комментарий (контроль A_1 A_2) к посылкам решаемой задачи. Здесь A_1 - то значение счетчика шагов интерпретатора ЛОСа, при достижении которого нужно обрабатывать отложенные фильтры. A_2 - набор установок на проверку отложенных фильтров, представляющих собой пятерки (B_1 B_2 B_3 B_4 B_5). B_1 - значение счетчика шагов интерпретатора, по достижении которого нужно обрабатывать данный фильтр (таким образом, A_1 - минимум значений B_1). B_2 - номер шага интерпретатора, на котором был создан отложенный фильтр; B_3 - логический символ, по которому следует предпринять обращение к справочнику "контроль", выполняющему проверку фильтра. B_4 - тройка (C_1 C_2 C_3), где (B_3, C_1, C_2) - стандартная ссылка на прием ГЕНОЛОГа, создавший программу справочника "контроль"; C_3 - набор дополнительных входных данных, необходимых для обработки фильтра. Заметим, что указанный прием ГЕНОЛОГа создает две программы - одна из них реализует основное действие, а другая является программой справочника "контроль", обслуживающей отложенную фильтрацию. B_5 - состояние текущей задачи до момент срабатывания приема, создавшего отложенный фильтр. К этому состоянию будет выполняться откат.

Чтобы создать в комментарии (контроль ...) установку на обработку отложенного фильтра, прием обращается к процедуре "установка(x1 x2 x3 x4 x5 x6 x7)". Здесь x1 - текущая задача; (x2,x3,x4) - ссылка на прием справочника "контроль", которому адресуется установка на обработку фильтра; x5 - набор дополнительных входных данных для справочника "контроль"; x6 - число шагов, по истечении которых требуется обработка фильтра; x7 - набор дополнительных комментариев, которыми снабжается копия текущей задачи, сохраняемая для отката. Программа этой процедуры несложна и реализует лишь непосредственную регистрацию в комментарии (контроль ...) своих входных данных.

Комментарий (контроль ...) анализируется при обращении к программам типов задач - "описать", "доказать", "преобразовать", "исследовать". Это происходит при каждом обращении, вне зависимости от значения текущего уровня. Роль диспетчера, организующего обращения к справочнику "контроль" по достижении требуемого числа шагов работы интерпретатора, играет процедура "Контроль(x1 x2)", где x1 - задача; x2 - комментарий (контроль ...) к ней. Эта же процедура изменяет при откате задачу x1 на ее старую версию. Ее программа несложна и разбираться здесь не будет.

При обращении к справочнику "контроль" на символе s используются следующие входные данные: значения переменных x1 - x5 передаются из сканирования задачи (x1 - текущая задача; x2-x4 равны 0); тройка (s,x6,x7) представляет собой ссылку на применяемый прием справочника "контроль"; x8 - набор дополнительных входных данных. Если откат не нужен, то справочник возвращает значение 1, иначе - значение 0. Заголовок приема справочника "контроль" имеет вид "контроль(A_1 A_2)", где A_1 - заголовок основного приема, обработкой отложенного фильтра которого занимается справочник; A_2 - символьный номер указателя "контроль(...)" в описании основного приема, задающего рассматриваемый отложенный фильтр. Информация о приеме справочника нужна потому, что из всей ветви его приемов, относящихся к данному символу s, сработать должен лишь тот прием, который реализует проверку выбранного отложенного фильтра. Соответственно, в начале каждой программы приема справочника "контроль" проверяется совпадение входных значений x6, x7 с "собственными" значениями приема.

2.2.5 Процедура "частичныйответ"

Если не требуется получить полный ответ задачи на описание, то при ее решении могут делаться неэквивалентные переходы, приводящие к рассмотрению частных подслучаев и нахождению частичных ответов. По мере возможности, приемы стараются не слишком сужать ситуацию. В тех случаях, когда этого недостаточно, задача снабжается целью "перечисление". Тогда найденный частичный ответ не выдается сразу, а сохраняется в комментарии (ответзадачи A), после чего продолжается рассмотрение альтернативных вариантов решения задачи. Решение обрывается по исчерпании всех средств, ограниченных максимальным уровнем, и выдается дизъюнкция найденных частичных ответов.

Для регистрации очередного частичного ответа в накопителе используется процедура "частичныйответ(x1 x2)". Здесь x1 - задача на описание; x2 - частичный ответ. Процедура не просто регистрирует в накопителе дизъюнктивные члены утверждения x2, но отбрасывает такие элементы накопителя, которые поглощаются дизъюнкцией остальных. Проверка поглощения происходит в быстром режиме, с помощью оператора "подборзначений".

2.3 Вспомогательные задачи

Чтобы создать вспомогательную задачу, а иногда и обратиться к ее решению, прием обычно использует некоторую вспомогательную процедуру. Перечислим наиболее часто используемые такие процедуры.

2.3.1 Процедура "спуск"

Если при решении задачи на описание либо на доказательство Z нужно предпринять неэквивалентный переход, заменив некоторое условие F на утверждение G, следствием которого оно является, то применяется операторное выражение "спуск(x1 x2 x3)". Здесь x1 - текущая задача; x2 - вхождение в список условий задачи x1 (для задач на доказательство - в саму задачу) утверждения F; x3 - утверждение G. Значением выражения является новая задача Z' того же типа, полученная заменой F на G. Выделение в отдельную процедуру процесса построения задачи Z' вызвано необходимостью коррекции различных ее компонент.

Случаи задач на описание и на доказательство рассматриваются отдельно. Если x1 - задача на описание, то создается копия x4 ее блока анализа. Комментарии в x4 переносятся из блока анализа задачи x1 с помощью справочника "спускоперандов". Даже если справочник не изменяет комментария, предпринимается его копирование, чтобы решение вспомогательной задачи не испортило задачи x1. Если задача x1 имела цель (известно ...), то задача x4 снабжается дополнительной целью "контроль". Последняя откладывает выдачу найденного ответа до уровня 3, чтобы на первых уровнях попытаться усмотреть нереализуемость ситуации. Она бывает необходима, например, при решении планиметрических задач на вычисление для отсечения нереализуемых вариантов. В x4 переносятся также комментарии, играющие роль "адресных структур" для идентифицирующих операторов. После создания задачи x4 вводится требуемая задача Z', присваиваемая переменной x5. Комментарии в нее переносятся из x1 справочником "спуск". Цель "упростить" у задачи x5 по умолчанию отбрасывается. При необходимости она должна быть восстановлена приемом, использующим процедуру "спуск".

Заметим, что у задач x4 и x5 копируются все те элементы, изменение которых могло бы испортить задачу x1. Последнее привело бы к неверному ответу, например, при разборе случаев, когда задача x1 порождает вспомогательные задачи многократно.

В случае задачи на доказательство действия аналогичны.

2.3.2 Процедура "преобразование"

Чтобы найти результат преобразования заданного выражения t относительно его контекста в текущей задаче Z, используется процедура "преобразование(x1 x2 x3 x4 x5 x6 x7)". Она вставляется в программу приема сканирования задачи компилятором ГЕНОЛОГа, например, при обработке нормализаторов, имеющих вид "задача(...)". Входные данные процедуры таковы. x1 - текущая задача Z; x2 - преобразуемое выражение t; x3 - набор утверждений, относительно которых преобразуется x2; x4 - список целей вспомогательной задачи Z'. К этому списку, возможно, добавляется элемент (уровеньобращения A), указывающий уровень A обращения к задаче Z'. Комментарии к задаче Z' извлекаются из задачи Z с помощью справочника "преобразование". По окончании решения задачи Z' этот же справочник обеспечивает отбор комментариев для возвращения их задаче Z. В списке x4 могут иметься элементы (комментарий K), которые дают не цели задачи Z', а комментарии K задачи Z'. Выходной переменной x5 присваивается ответ задачи Z'; переменной x6 - набор использованных посылок списка x3; переменной x7 - задача Z'. Последнее позволяет извлечь из Z' дополнительные сведения, полезные при продолжении решения.

Обращение к справочнику "преобразование" состоит из следующих данных: x1

- x4 - те же, что у процедуры "преобразование(...)"; x5 - комментарий, заголовком которого служит текущий логический символ; x6 - либо 0 (перенесение комментария из Z в Z'), либо задача Z' (обратное перенесение). Если перенесение комментария целесообразно, справочник возвращает его модифицированную версию, иначе - выдает 0.

Программа процедуры "преобразование" создает вспомогательную задачу Z', присваивая ее переменной x8. Задаче придается дополнительная цель "преобразование", являющаяся указателем на источник задачи. Веса посылок задачи x8 полагаются равными 20, т.е. сканирование посылок блокируется. Это делается из-за того, что обработка контекста преобразования уже должна была произойти в задаче x1. Однако, при отсутствии в наборе x4 цели "извлекается" находятся все посылки задачи x1, веса которых меньше 2, и эти их веса передаются в задачу x8. Таким образом обеспечивается разблокировка рассмотрения новых для задачи x1 посылок. Далее - переход через "ветвь 2".

Некоторые вспомогательные задачи x8 могут создаваться различными приемами на коротком промежутке времени и независимо друг от друга. Чтобы избежать повторных попыток их решения, используется специальный буфер. Его роль играет комментарий (преобразование A) к посылкам исходной задачи. В наборе A накапливаются тройки (B_1 B_2 B_3), где B_1 - исходная версия задачи x8; B_2 - версия этой задачи на момент получения ее ответа; B_3 - ответ. Указателем на использование буфера служит логический символ "буфер", помещаемый при обращении в набор x4.

Вводится накопитель x9 результата решения задачи x8. Если имеется указание на использование буфера, причем в буфере находится готовый ответ, то он передается переменной x9, причем x8 заменяется на извлеченную из буфера версию этой задачи, возникающую на момент получения ответа. В противном случае - переход через "ветвь 2".

Если в x4 имеется указатель "буфер", то переменной x10 присваивается копия задачи x8, сохраняемая для последующей регситрации в буфере. Уровень обращения к задаче x8 переустанавливается согласно элементу (уровеньобращения N) из x4. При отсутствии такого элемента уровень обращения берется равным текущему уровню сканирования текущей задачи. Этой ситуации лучше избегать; компилятор ГЕНОЛОГа создает обращения к данной процедуре, явно указывая уровень обращения. Если нужно ввести ограничение на трудоемкость решения задачи x8, то в список x4 заносится элемент "лимит(M)". Тогда перед обращением к решению вводится установка на обрыв по достижении M шагов. В программе независимо рассматриваются оба случая - с ограничением и без него. В каждом из них переменной x9 переприсваивается найденный ответ; если имелся указатель "буфер", то предпринимается регистрация результата обращения в буфере. Далее - откат к переходу через оператор "ветвь 1", расположенный после инициализации переменной x9.

Здесь проверяется, что x9 отлично от символов "противоречие" и "отказ". Происходит регистрация в задаче x1 данных об использованных посылках и перенесение в нее из задачи x8 комментариев, отбираемых справочником "преобразование". Затем выдается результат.

2.3.3 Процедура "вспомописание"

Если нормализатор приема ГЕНОЛОГа выполняет обработку терма t с помощью задачи на описание, то компилятор использует процедуру "вспомописание(x1 x2 x3 x4 x5 x6 x7)", аналогичную рассмотренной выше процедуре "преобразование". Как и ранее, x1 - текущая задача Z; x2 - преобразуемое утверждение t; x3 - набор утверждений, составляющих контекст преобразования; x4 - список целей, пополненный дополнительными элементами. Процедура создает вспомогательную задачу на описание Z', условиями которой являются все конъюнктивные члены утверждения t, а также необходимые сопровождающие утверждения, извлекаемые из x3. Посылками задачи Z' служат все не зависящие от ее неизвестных утверждения списка x3. Выходной переменной x5 присваивается ответ задачи Z'; переменной x6 - список использованных посылок; переменной x7 - задача Z'. Комментарии из Z в Z' и обратно пересылаются так же, как и выше - с помощью справочника "преобразование". В списке x4 могут встречаться дополнительные элементы следующих типов:

1. (вспомописание X A). Элемент означает, что задача решается относительно неизвестных списка X, введенных как вспомогательные обозначения выражений набора A. Утверждение t перед обращением к процедуре уже переформулировано в терминах переменных X, причем после получения ответа R переменные X исключаются из него путем подстановки их "значений"A;
2. (уровеньобращения A). Элемент указывает максимальный уровень задачи Z';
3. (комментарии A). Элемент указывает список A комментариев, передаваемых задаче Z';
4. "вход". Элемент означает, что блок анализа у задачи Z' не вводится;
5. "контекст". Элемент означает, что в список условий задачи Z', кроме конъюнктивных членов утверждения t, включаются все содержащие неизвестные задачи Z' утверждения из набора x3. При отсутствии данного элемента в список условий переносятся лишь избранные неизвестные утверждения списка x3.
6. "пересмотр". Элемент означает, что веса посылок задачи Z' изначально берутся равными 0.
7. "буфер". Элемент означает, что используется буфер результатов обращения к процедуре "вспомописание". Он аналогичен буферу процедуры "преобразование", но вместо комментария (преобразование ...) берется комментарий (вспомописание ...).

Переходим к рассмотрению программы процедуры. Если в x4 имеется элемент (вспомописание X A), то список x4 пополняется элементом (неизвестные X). Предполагается, что до этого в x4 не было цели, перечисляющей неизвестные.

Определяется список посылок x9 вспомогательной задачи - он образован всеми утверждениями набора x3, не содержащими неизвестных новой задачи. Создается новая задача Z', присваиваемая переменной x11. Первоначально ее условиями служат конъюнктивные члены утверждения x2. Веса посылок (при отсутствии элемента "пересмотр") берутся равными 20, и реализуется частичная разблокировка посылок, аналогичная случаю задач на преобразование.

Просматриваются утверждения списка x3, содержащие неизвестные задачи Z'. Если в x4 имеется элемент "контекст" либо утверждение задает тип значения переменной, то оно заносится в список условий задачи Z'. При наличии в x4 элемента (вспомописание X A) предпринимается попытка усмотреть типы значений переменных X и тоже зарегистрировать их в списке условий новой задачи.

Просматриваются условия новой задачи и находятся их подвыражения с неизвестными, требующие сопровождения по о.д.з. Если сопровождающее утверждение U содержит неизвестные и не извлекается из списка условий, то предпринимается попытка убедиться в том, что оно вытекает из x3. При наличии элемента (вспомописание ...) предварительно предпринимается расшифровка водящих в U вспомогательных обозначений. Затем U и использованные для его обоснования элементы x3, не являющиеся посылками задачи Z', добавляются к списку условий.

Просматриваются содержащие неизвестные новой задачи элементы списка x3, и выполняются обращения к справочнику "вспомописание", принимающему решение о целесообразности добавления их к списку условий.

Дальнейшая схема действий аналогична случаю задач на преобразование. Сначала предпринимается попытка усмотреть ответ из буфера. Если она не удается, то предпринимается обращение к решению задачи Z', с учетом имеющегося ограничителя трудоемкости. Результат заносится в буфер. После нахождения ответа x13 проверяется наличие элемента (вспомописание X A) и предпринимается подстановка в этот ответ выражений A вместо переменных X. В задачу Z переносится информация об использованных посылках, а также переносятся комментарии из Z', отбираемые справочником "преобразование". Наконец, все посылки задачи Z', помеченные комментарием "облвхожд", переносятся в блок анализа задачи Z. Если в этом случае блока анализа не было, то он вводится.

2.3.4 Процедуры "извлекается", "выводимо", "сильноизвлекается"

Для быстрой проверки истинности утверждений в контексте сканирования текущей задачи могут использоваться процедуры "извлекается(x1 x2 x3)" и "сильноизвлекается(x1 x2 x3)". Здесь x1 - проверяемое утверждение; x2 - список утверждений, образующих контекст проверки. Решается вспомогательная задача на доказательство с условием x1 и посылками x2. Если на нее получается ответ "истина", то переменной x3 присваивается список использованных посылок.

В случае процедуры "извлекается" веса посылок равны 20, а задача решается с максимальным уровнем 4. В случае процедуры "выводимо" веса посылок нулевые, а максимальный уровень либо равен текущему уровню текущей задачи, либо специально устанавливается перед обращением. В случае процедуры "сильноизвлекается" веса посылок нулевые, а максимальный уровень равен 7. Все эти процедуры в приемах ГЕНОЛОГа не используются; в прочих ситуациях чаще всего встречаются обращения к процедуре "извлекается".

2.3.5 Процедура "следствие"

Для проверки истинности вспомогательных утверждений приемы ГЕНОЛОГа используют процедуру "следствие(x1 x2 x3 x4)". Здесь x1 - проверяемое утверждение; x2 - набор утверждений, образующих контекст проверки; x3 - набор комментариев вспомогательной задачи на доказательство Z', создаваемой для проверки x1. Максимальный уровень задачи Z' устанавливается перед обращением к процедуре. Если на задачу Z' получен ответ "истина", то переменной x4 присваивается список использованных утверждений набора x2.

Если x1 имеет вид кванторной импликации, то все антецеденты этой импликации будут присоединены к списку посылок задачи Z', а условием задачи станет консеквент. В этом случае предусмотрена возможность сопровождать антецеденты комментариями. Ее обеспечивает элемент (комментарийпосылки A_1 A_2), заносимый в набор x3. Здесь A_1 - тот антецедент, к которому относится комментарий A_2.

Если набор x3 имеет элемент "извлекается", то веса посылок задачи Z' полагаются равными 20, иначе они равны 0.

2.3.6 Процедура "определениепараметра"

Если в посылках задачи встречается числовое равенство, то иногда бывает полезно выразить с его помощью один из параметров через другие. Так как равенство может иметь несколько числовых параметров, то предварительно нужно определить, относительно какого из них равенство будет разрешаться. Для выбора параметра и реализации попытки разрешения используется процедура "определениепараметра(x1 x2 x3 x4 x5)". Здесь x1 - рассматриваемое равенство; x2 - набор утверждений, составляющих внешний контекст; x3 - набор дополнительных установок на разрешение. Параметр, относительно которого происходит разрешение - не обязательно переменная; это может быть и некоторое атомарное составное неконстантное числовое выражение, т.е. выражение, имеющее нечисловой операнд. Если удалось разрешить равенство x1 относительно некоторого такого числового атома t, то переменной x4 присваивается результат разрешения; переменной x5 - список использованных посылок списка x2. Если в наборе x2 имеется символ "одз", то в x5 могут быть занесены некоторые дополнительные утверждения, необходимые для сопровождения по о.д.з.

Элемент (уровень N) в наборе x3 указывает на максимально допустимую эвристическую оценку N сложности разрешения (см. ниже). Она позволяет избежать появления громоздких выражений. Элемент (неизвестные A) указывает список переменных A, являющихся приоритетными при разрешении. Если хотя бы одна из них входит в x1, то не указанные в A числовые атомы не рассматриваются.

Перейдем к рассмотрению программы процедуры. Прежде всего, проверяется наличие в x2 указателя "одз". Если он есть, то удаляется из x2, причем в этом случае переменной x6 присваивается 1, иначе x6 полагается равным 0. Переменной x7 присваивается список всех числовых атомов равенства x1, не расположенных внутри термов "значение(...)". Это - кандидаты на разрешение. При наличии в x3 элемента (неизвестные ...) выполняется соответствующая коррекция списка x7. Если x1 уже имеет вид равенства терма списка x7 выражению, не содержащему этого терма, то разрешение не требуется, и процедура выдает значение "ложь". Иначе, при непустом x7, начинается цикл составления списка x8 пар (числовой атом - оценка его приоритета). Пары упорядочены по возрастанию оценок (т.е. больше оценка - меньше приоритет). При определении оценки приоритета N текущего атома x9 рассматриваются следующие случаи:

1. Единственное вхождение x9 в x1 расположено только внутри операций "плюс", "минус". Тогда $N = 1$.

2. Единственное вхождение x9 в x1 расположено только внутри операций "плюс", "минус", "умножение", причем все коэффициенты не обращаются в 0. Тогда $N = 2$. Если условие на коэффициенты не выполнено, то $N = 3$.

3. Единственное вхождение x9 в x1 расположено только внутри операций "плюс", "минус", "умножение", "степень". Если все коэффициенты отличны от нуля, показатели степени над x9 - рациональные с нечетным числителем, то $N = 4$;

если условие на коэффициенты не выполнено, а условие на степени - выполнено, то $N = 5$; если оба этих условия не выполнены, то $N = 6$.

4. Единственное вхождение x9 в x1 расположено только внутри операций "минус", "умножение", "дробь", причем все коэффициенты не равны 0. Тогда $N = 5$. Если последнее условие не выполнено, то $N = 6$.
5. Единственное вхождение x9 в x1 расположено только внутри операций "плюс", "минус", "дробь", "умножение", "степень". Тогда $N = 8$.
6. В прочих случаях, если x9 имеет единственное вхождение в x1, $N = 10$.
7. Если целочисленный числовой атом x9 имеет более одного вхождения в x1, причем не все они расположены только под операциями "плюс", "минус", то $N = 50$.
8. Если x9 - переменная, относительно которой x1 линейно, то $N = 9$.
9. В остальных случаях, если число вхождений x9 в x1 равно 2, то $N = 15$, иначе $N = 20$.

По завершении составления списка x8 начинается последовательный просмотр пар (t,N). Он обрывается по достижении значения N, превышающего установленный указателем (уровень ...) лимит. Переменной x10 присваивается выражение t; переменной x11 - список всех не содержащих t утверждений списка x2. Он будет играть роль список посылок вспомогательной задачи на описание Z. Если t - переменная, то она будет неизвестной x12 задачи Z. Иначе выбирается новая переменная x12. Чтобы получить условие задачи, в первом случае берется само равенство x1, во втором - результат замены в нем всех вхождений t на x12. Результат присваивается переменной x13. Вспомогательная задача присваивается переменной x14; она имеет цели "полный", "прямойответ", "явное".

Перед обращением к решению задачи x14 предпринимается пополнение списка ее условий сопровождающими утверждениями. Накопителем этих утверждений является набор x15. Сначала в него заносятся утверждения, необходимые для сопровождения по о.д.з. Если индикатор x6 равен 0, то перед занесением утверждения проверяется, что оно является следствием утверждений списка x2. Затем предпринимаются попытки усмотреть знак выражения t (нестрогий либо строгий), и к x15 присоединяется соответствующее неравенство. Утверждения списка x15 обрабатываются нормализатором "нормодз", после чего заносятся в список условий. Задача x14 решается с максимальным уровнем 6. Перед выдачей ответа реализуется подстановка в него выражения t вместо неизвестной x12.

2.4 Пакетные анализаторы: обновленная версия

Перед тем, как перейти к описанию процедур, используемых в пакетных операторах ГЕНОЛО́Га, вернемся к общему описанию устройства пакетных анализаторов. Интенсивное использование их в планиметрии потребовало модернизации старой, чересчур громоздкой и неудобной архитектуры, которая была описана в первом томе. Приведем здесь краткое описание обновленной версии.

2.4.1 Функционирование анализатора

Пакетным аналогом задачи на исследование является пакетный оператор, названный анализатором. Он обрабатывает копию списка посылок текущей задачи Z, оформленную в виде вспомогательной задачи на исследование Z'. Анализатор не обращается для сканирования Z' к основной базе приемов, а использует свои собственные приемы. В остальном его поведение практически ничем не отличается от обычного процесса решения задачи на исследование. Так как приемов анализатора существенно меньше, чем в основной базе, то удается за сравнительно небольшое время получить значительное число следствий. Иэ этих следствий отбираются лишь ценные для внешней задачи Z, которые по окончании работы анализатора переносятся в ее список посылок. Отбор выполняется приемами анализатора, помечающими нужные посылки комментарием "внешвывод". Обращение к анализатору обеспечивается специальными приемами, указывающими лимит времени его работы. Передача отобранных утверждений происходит по исчерпании этого лимита. При получении особо важных следствий возможен немедленный обрыв работы. Входными данными обращения к анализатору служат: задача Z, максимальный уровень используемых приемов и целевая установка, передаваемая задаче Z'. Вспомогательная задача Z' создается самим анализатором. Он же реализует передачу отобранных утверждений в задачу Z.

2.4.2 Запись приема анализатора на ГЕНОЛО́Ге

Прием анализатора имеет теорему вида $\forall x(A_1 \ \& \ \ldots \ \& \ A_n \to A_0)$, при помощи которой осуществляется вывод следствия A_0. Заголовок приема - "внутрвывод(B)", где B - название анализатора. Если прием не выводит следствие, а выполняет тождественную либо эквивалентную замену, то A_0, соответственно, является равенством либо эквивалентностью. Замена всегда выполняется слева направо. Приемы, выполняющие замену, выделяются указателем "внутрпреобр". Способ обработки антецедентов $A_i (i = 1, \ldots, n)$ обычным образом уточняется указателями приема.

Если прием анализатора выводит настолько ценное следствие, что целесообразно безотлагательное перенесение его во внешнюю задачу и возвращение к сканированию этой задачи, то используется указатель "обрыв" либо "обрыв(A)", где A - фильтр, уточняющий условие обрыва работы анализатора. В случае обрыва сформированное приемом утверждение автоматически снабжается комментарием "внешвывод" и таким образом переносится во внешнюю задачу.

2.4.3 Обращение к анализатору

Обращение к пакетному анализатору A выполняется приемом сканирования задачи, имеющим заголовок "замечание". Прием снабжается указателем "анализатор(A B уровень(u))". Здесь u - максимальный уровень срабатывания приемов анализатора, допустимых при его работе, B - список целей вспомогательной задачи анализатора. Неизвестные текущей задачи передаются вспомогательной задаче автоматически, и их можно в списке B не указывать. Кроме того, через список B можно задавать ограничитель трудоемкости обращения к анализатору - терм "лимит(N)", где N - допустимый лимит шагов работы интерпретатора ЛОСа. Этот терм в цели вспомогательной задачи анализатора не включается.

2.5 Процедуры пакетных операторов

В программах пакетных операторов используется ряд процедур общего назначения. Они обеспечивают инициализацию работы оператора; пошаговую трассировку; дополнительную фильтрацию из общих соображений; разбор случаев; взаимодействие с буферами; обращение к вспомогательным задачам и т.п. Перечислим наиболее часто используемые такие процедуры.

2.5.1 Процедура "контрольнормализации"

Многие пакетные нормализаторы используют специальные буферы, в которых сохраняются результаты предыдущих 50 обращений. Если буфер содержит готовый результат, то он сразу выдается нормализатором. Иначе - начинаются вычисления, и по завершении их результат передается в буфер. Буферы очень эффективны в тех случаях, когда имеется несколько различных похожих приемов, обращающихся к нормализации одних и тех же термов. Они избавляют от необходимости специальной склейки таких приемов в общую алгоритмическую конструкцию для экономии обращений. Если хотя бы один прием обратился к нормализатору, то последущие идентичные обращения оказываются мгновенными. Роль буфера играет комментарий (контрольнормализации $f\ A\ v$), где f - название нормализатора; A - набор длины 50, элементами которого служат нули либо наборы $(1\ t\ P\ K\ r\ Q)$ - результаты предыдущих обращений. v - текущая позиция набора A, на которую будет заноситься очередной результат. Она сдвигается циклическим образом. Наличие элемента 1 в начале набора объясняется лишь особенностями оператора "биключ", используемого для извлечения нужных данных. t - терм, поступивший на вход нормализатора; P - список посылок, относительно которого выполнялись преобразования; K - подмножество списка комментариев, отбираемое для идентификации обращений (различия в прочих комментариях будут игнорироваться); r - результат нормализации; Q - список фактически использованных посылок.

Чтобы пакетный нормализатор использовал буфер, в описании его формата должен иметься символ "контрольнормализации". Обычно нормализаторы общей стандартизации, выполняющие сравнительно "дешевые" преобразования, не используют буфера. В более трудоемких процедурах (например, "разложитьнамножители") его целесообразно использовать.

Программа оператора, использующего буфер, начинается с обращения к процедуре "контрольнормализации(x1 x2 x3 x4 x5 x6)". Здесь x1, x2, x3 - входные данные обращения к нормализатору, т.е. преобразуемый терм, список посылок и список комментариев. x4 - название нормализатора. Процедура проверяет наличие в буфере готового результата. Если он есть, то присваивается выходной переменной x5. Переменной x6 при этом присваивается 0. Если готового результата нет, то переменной x6 присваивается тройка (x1, x2, K), необходимая для последующей регистрации результата в буфере. Здесь K - указанное выше подмножество комментариев, используемое для идентификации обращения. Данная тройка будет сохраняться на все время выполнения работы нормализатора, а перед завершением работы процедура "учетнормализации" использует ее для регистрации результата в буфере. Если переменной x6 присвоена тройка, то переменной x5 присваивается x1 или результат частичного преобразования x1. Последний берется из того же буфера - если имелся результат преобразований для подмножества посылок x2. В этом случае преобразования будут продолжены начиная сразу с терма x5.

Кроме учета буфера, процедура "контрольнормализации" выполняет ряд других функций, которые оказалось удобным передать ей из-за ее размещения в самом начале программы пакета. Например, она участвует в отображении на экране обращения к нормализатору при трассировке решения.

Выдача информации об обращении к нормализатору выполняется процедурой отладчика ЛОСа. При наличии подходящего режима трассировки, интерпретатор усматривает обращение к процедуре "контрольнормализации" и обращается к отладчику. Последний анализирует ситуацию и выдает на экран необходимые тексты. Все это делается без участия программы "контрольнормализации". Она начинает выполняться уже после выхода из кадра трассировки. Переходим к рассмотрению этой программы.

Прежде всего, выключается счетчик шагов интерпретатора и предпринимается ряд действий по расчистке комментариев, обеспечивающих трассировку. Например, из списка комментариев удаляется уже использованный элемент (титр ...), который в противном случае мог бы испортить отображение трассировки в подпроцедурах. После расчистки - откат к переходу через "ветвь 2", где счетчик шагов снова включается. Отключение счетчика шагов предпринимается, чтобы остановка интерпретатора на шаге решения задачи с заданным номером не зависела от применяемого режима трассировки.

Элемент "повтор" в x3 определяет контроль возможного зацикливания при рекурсивных обращениях. Если имеется внешнее обращение к тому же самому нормализатору для обработки того же самого терма x1, то цикл обращений разрывается - выдается неизмененное входное данное x1.

При наличии в x3 элемента (разборслучаев ...) предпринимается регистрация в нем пары (комментарий "коррекцияпосылок" - исходный список посылок). Режим нормализации с разбором случаев понадобился при вычислении пределов выражений с параметрами. Подробнее о нем см. ниже.

Перед просмотром буфера инициализируются значения переменных x9, x10. Первой будет переприсвоен тот элемент буфера, который определит частичное преобразование терма x1. Вторая представляет собой индикатор уровня средств нормализатора, подключаемых к обработке терма x1 согласно комментариям обращения. Чем больше этот индикатор, тем уже группа применяемых приемов. Лишь немногие нормализаторы имеют шкалу уровней средств, и для них процедура "контрольнормализации" указывает конкретные значения x10. Не вдаваясь в подробности, относящиеся к таким нормализаторам, ограничимся перечислением их названий: "видумножение", "уравнменьшеилиравно", "уравнменьше", "стандменьше", "стандменьшеилиравно", "стандплюс", "нормпредел", "асимптоценка". Все это - большие или даже очень большие пакеты.

Если x3 содержит комментарий "новый", то попытка извлечения результата из буфера блокируется. В этой ситуации нормализатор будет выполнять все преобразования заново. Иначе находится комментарий (контрольнормализации x4 A v) к посылкам исходной задачи, и переменной x13 присваивается набор A результатов предыдущих обращений. В нем просматриваются элементы E вида (1 x1 P K r Q), соответствующие обрабатываемому терму x1.

Если список посылок P равен списку посылок x2 текущего обращения, то в наборе K рассматривается элемент (ключ q), сохраняющий индикатор q уровня применявшихся средств (т.е. указанное выше значение x10). Если q отличается от x10, причем либо результат r отличен от x1, либо x10 меньше q, либо x10 больше 9, то элемент E игнорируется. Иначе этот элемент используется как указатель на результат r текущего обращения. Перед выдачей результата корректируются комментарии нормализатора (выводимо ...), (коррекцияпосылок ...). Вся необходимая для их коррекции информация имеется в списках K и Q.

Если список посылок x2 - собственное подмножество P, то рассматривается элемент (ключ q) набора K. Если x1 отлично от r, либо x10 меньше q, либо x10 больше 9, то элемент E игнорируется. В противном случае терм x1 не был изменен при усиленной обработке относительно расширенного списка посылок. Поэтому можно считать, что и при данном обращении он не изменится, и выдать его в качестве результата.

Если для текущего элемента E ни одна из указанных выше ситуаций не имеет места, то рассматривается элемент (ключ q) набора K. Если q меньше x10 либо x10 больше 9, то элемент E игнорируется. Это же происходит, если K имеет элементы (рядтейлора ...), (Полюс ...), не входящие в x3. Такие элементы содержат дополнительную входную логическую информацию (например, точку, для которой выполняется разложение в ряд). В прочих случаях проверяется, что старый список посылок P включается в x2, и переменной x9 переприсваивается набор E. Он определит частичное преобразование терма x1. После переприсвоения просмотр элементов E продолжается, так как может найтись другой элемент, дающий готовый результат обращения.

Если по окончании просмотра оказалось, что x9 не равно 0, то x1 заменяется на новое значение r и выполняется коррекция комментария (выводимо ...): учитываются утверждения, использованные при получении промежуточного результата r. Затем вводится накопитель x12 выходного значения K. В него заносится сначала единственный элемент (ключ x10).

Если в наборе x3 имелся элемент (буфер A), то при $A = 1$ в x12 заносится элемент "стоп". Такой элемент будет блокировать занесение в буфер результата данного обращения. Если же $A \neq 1$, то оно заменяется на 1. Тогда окажется заблокированной регистрация в буфере всех результатов обращения к вспомогательным нормализаторам, которыми будет пользоваться данный нормализатор. В x12 переносятся из x3 следующие элементы: (выводимо ...), (коррекцияпосылок ...), (рядтейлора ...), (Полюс ...). Перед выдачей результата еще раз происходит контроль зацикливания по цепочке рекурсивных обращений, на этот раз для измененного терма x1.

2.5.2 Процедура "учетнормализации"

Эта процедура используется приемами нормализаторов для регистрации в буфере результата преобразований. Обращение к ней имеет вид "учетнормализации(x1 x2 x3 x4)", где x1 - результат; x2 - название нормализатора; x3 - список комментариев нормализатора; x4 - тройка (A_1 A_2 K), сформированная процедурой "контрольнормализации" для идентификации обращения в буфере. Именно, A_1 - исходный вид преобразуемого терма; A_2 - список посылок; K - дополнительные данные.

Прежде всего, программа проверяет необходимость регистрации результата. Регистрация блокируется, если в K имеется элемент "стоп", либо в x3 имеется элемент "откат". Это же происходит в некоторых других особых случаях. Перед регистрацией модифицируются комментарии (выводимо ...) и (коррекцияпосылок ...).

2.5.3 Процедура "контрользамены"

Процедура используется нормализаторами, главным образом, для двух целей. Во-первых, чтобы получить возможность заблокировать срабатывание приема из каких-либо общих соображений, например, при тестировании системы вывода теорем. Во-вторых, чтобы зарегистрировать в архиве факт срабатывания приема. Отображение на экране текущего преобразования, выполняемого нормализатором, с данной процедурой не связано - оно инициируется при попытке изменения значения программной переменной x1.

Обращение к процедуре имеет вид "контрользамены(x1 x2 x3 x4 x5 x6)". Здесь x1 - преобразуемый терм; x2 - заменяющий терм либо пара (символ "выход" - заменяющий терм), если прием обрывает применение нормализатора; x3 - список посылок; x4 - список комментариев нормализатора; x5 - набор (символ "прием" - A_1 - A_2 - A_3), ссылающийся на примененный прием. Как и обычно, A_1 и A_2 суть логический символ и номер узла статьи этого символа в базе приемов. Однако, A_3 - не заголовок приема, а лишь указатель направления замены. x6 - название нормализатора. Если выполнение замены нецелесообразно, то оператор ложен.

Программа процедуры проверяет различие заменяемого и заменяющего термов. Блокировка замены реализуется при наличии комментария (приемы ...) к посылкам исходной задачи, отменяющего срабатывания заданных приемов. Учет факта срабатывания приема в архиве выполняется уже рассмотренной выше процедурой "учетприменения".

2.5.4 Процедура "контрольбуфера"

Эта процедура аналогична процедуре "контрольнормализации", однако применяется в пакетных проверочных операторах и пакетных синтезаторах. Здесь роль буфера, сохраняющего результаты 30 последних обращений к пакету, играет комментарий (контрольбуфера A_1 A_2 A_3) к посылкам исходной задачи. A_1 - название пакета; A_2 - набор четверок (B_1 B_2 B_3 B_4), где B_1 - набор входных данных пакета с пропущенным списком посылок; B_2 - список посылок; B_4 - результат обращения, т.е. набор значений выходных переменных пакета в случае позитивного результата и 0 в случае негативного. Набор A_2 имеет 30 позиций, причем A_3 - вхождение текущей позиции, на которой будет регистироваться очередной результат.

Обращение к процедуре вставляется во все пакеты указанных типов, безотносительно к их формату. Оно имеет вид "контрольбуфера(x1 x2 x3 x4)". Здесь x1 - название пакетного оператора. Для перечисляющиего синтезатора x1 имеет вид терма "f(перечисление)", где f - название синтезатора. x2 - набор входных данных пакета. Если в буфере имеется готовый результат, то в случае позитивного результата происходит присвоение его переменной x3, а переменной x4 присваивается 0. В случае негативного результата процедура выдает "ложь". Если в буфере нет готового результата, то переменной x3 присваивается 0, а переменной x4 - модифицированный для последующей регистрации в буфере набор x2. Если будет получен позитивный результат, то регистрацию выполнит процедура "учетвбуфере". Иначе, после исчерпания всех попыток, произойдет откат к процедуре "контрольбуфера", так как она является перечисляющей. При этом будет реализована регистрация в буфере негативного результата.

Отображение на экране обращения к проверочному оператору либо синтезатору происходит при обращении к процедуре "контрольбуфера". Как и в случае нормализаторов, сама программа в таком отображении не участвует - просто при попытке обращения к ней интерпретатор ЛОСа передает управление отладчику ЛОСа, который и реализует все необходимые действия по организации кадра трассировки. Уже после работы в этом кадре начинается выполнение программы процедуры. Прежде всего, при выключенном счетчике шагов, реализуется коррекция комментариев, используемых при трассировке. Затем - откат к переходу через "ветвь 2".

Если в комментариях обращения имелся элемент (лимит N), причем число на счетчике шагов интерпретатора ЛОСа больше N, то сразу предпринимается выход по "ложь". Этот способ обрывать работу пакетов используется редко - обычно откат по исчерпании разрешенной трудоемкости реализуется оператором "лимит(...)".

Переменной x5 присваивается вхождение в набор x2 списка посылок обращения. Если все входные параметры обращения суть константные термы, то по вхождению x5 заносится символ "пустоеслово" - список посылок здесь не нужен. Далее, при непустом списке посылок, он заменяется в наборе x2 на свою копию. Это делается, чтобы последующая работа пакета не изменила информации о первоначальном виде входных данных.

Переменной x7 присваивается тот комментарий (контрольбуфера ...), который является буфером результатов обращения к данному пакету. В нем ищется четверка x9, соответствующая текущему набору входных данных (с пропущенным списком посылок). Если результат обращения позитивный, то проверяется, что список посылок четверки x9 включается в список посылок обращения. После этого выдается готовый результат. Если результат негативный, то проверяется, что наоборот, текущий список посылок включается в список посылок четверки x9. После этого - выход по "ложь". Если в буфере результат не найден, то выдаются значения x3 = 0; x4 = x2, и далее реализуется база приемов пакета. Если действия пакета безрезультатны, то при откате происходит возвращение в процедуру "контрольбуфера", и она регистрирует негативный результат. Особо учитывается случай перечисляющих пакетов. Здесь регистрация негативного результата не предпринимается, так как откат может происходить после цепочки позитивных реализаций пакета - по исчерпании перечисления.

2.5.5 Процедура "учетвбуфере"

Процедура используется для регистрации в буфере результатов обращения к проверочному оператору либо синтезатору. Обращение к ней имеет вид "учетвбуфере(x1 x2 x3)". x1 - название оператора либо (в случае перечисляющего синтезатора) терм "f(перечисление)", где f - название оператора. x2 - набор входных данных обращения к оператору, модифицированный процедурой "контрольбуфера". x3 - результат обращения (набор значений выходных переменных либо значение единственной такой переменной).

Предпринимается регистрация результата обращения в буфере. Если обращение к пакету выполнялось с комментарием "блок", то срабатывания приемов, отличных от непосредственного усмотрения результата из посылок, блокируются. Это происходит внутри данной процедуры - при невырожденном результате x3 реализуется немедленный (без регистрации в буфере) выход по значению "ложь".

2.5.6 Процедуры "учетвывода" и "коррекцияпосылок"

Если нормализатор преобразует выражение таким образом, что возникает необходимость в новых сопровождающих по о.д.з. утверждениях, то эти утверждения регистрируются в комментарии (коррекцияпосылок ...), а затем оттуда переносятся в посылки нормализатора. Первое делается процедурой "учетвывода"; второе - оператором программы приема, заменяющим текущий список посылок на значение операторного выражения "коррекцияпосылок(...)". Комментарии (коррекцияпосылок ...) будут необходимы для передачи информации о сопровождении по о.д.з. внешним процедурам, обратившимся к нормализатору.

2.5.7 Процедура "буфер"

Процедура расчищает буферы результатов обращения к пакетным операторам и вспомогательным задачам на преобразование, решавшимся при реализации пакета. Обращение к ней имеет вид "буфер(0)". Используется расчистка буферов пакетных операторов крайне редко. Например, она может быть полезна при трассировке - если запускается повтор решения вспомогательной задачи, то нужно предотвратить влияние на него информации, сохраненной в буферах.

2.5.8 Разбор случаев в нормализаторах

Если в формате нормализатора имеется указатель "разборслучаев", то он может работать в режиме разбора случаев по условиям на параметры входных термов. Дизъюнкции, по которым должны разбираться случаи, определяются в процессе преобразований. Если такая дизъюнкция возникла, то предпринимается откат к началу преобразований и проведение их для каждого подслучая по отдельности. Результаты сохраняются в специальном накопителе, роль которого играет комментарий (разборслучаев ...). По завершении разбора случаев составляется условное выражение. Оно упрощается и выдается в качестве результата обработки. Результаты вычислений до отката сохраняются в буферах пакетных операторов, так что после отката уже пройденный участок быстро повторяется. Возможно рассмотрение вложенных разборов случаев; схема их зарегистрирована в комментарии (разборслучаев A_1 A_2 A_3). Этот комментарий должен присутствовать в обращении к оператору, причем изначально $A_1 = A_2 = 0$; $A_3 =$ "пустоеслово".

В процессе работы нормализатора могут срабатывать приемы, инициирующие разбор случаев в текущем анализируемом подслучае. Если нормализатор имел рекурсивные обращения к самому себе или другим разбирающим случаи нормализаторам, то прием, инициирующий разбор случаев, может сработать на некоторой глубине цепочки рекурсивных обращений. В этом случае откат произойдет по всей цепочке обращений, с выходом на корневое обращение. Технически он реализован как немедленная выдача ответа - терма "разборслучаев(или($P_1 \ldots P_n$))", указывающего список альтернатив $P_1, \ldots, P_n$. Признаком, указывающим на корневое обращение, является отсутствие комментария "подчинено", который вводился для вспомогательных обращений.

Вернемся к элементам A_1, A_2, A_3 комментария (разборслучаев ...). Первый из них - сигнализатор о наличии в комментарии только что зарегистрированной новой серии подслучаев, инициирующей очередной разбор. Он равен 1, если такая серия есть, иначе - равен 0. A_2 - пара (исходный комментарий (коррекцияпосылок ...) - исходный список посылок). Набор A_3 представляет собой накопитель подслучаев - наборов ($B_1 \ldots B_6$). Группа этих наборов, соответствующих очередной серии подслучаев текущего случая, регистрируется в начале набора. В ней альтернативы $P_1, \ldots, P_n$ перечисляются в обратном порядке.

B_1 есть список посылок подслучая, содержащий, в том числе, определяющее подслучай утверждение B_2. B_6 - преобразуемый в данном подслучае терм. Если B_2 имело вид равенства, явно задающего значение некоторого параметра, то B_6 может отличаться от исходного терма - вследствие подстановки в него данного значения и упрощения. В этой ситуации параметр исключается и из посылок B_1, куда B_2 уже не заносится. Если имела место подстановка значения параметра, то B_3 - набор той же длины, что B_1, указывающий для каждого утверждения списка B_1 набор исходных посылок, следствием которых оно является. Иначе B_3 есть 0. B_4 - набор комментариев нормализатора, используемых в подслучае. B_5 - сначала 0, а затем результат нормализации предыдущих альтернатив той же серии подслучаев. Если таких альтернатив было несколько, то результат собирается из фрагментов как условное выражение, которое сразу же упрощается. Отличие значения B_5 от нуля (равенство его символу 1 либо результату нормализации уже обработанных альтернатив серии подслучаев) является указателем на границу серии подслучаев. Предшествующий набор будет относиться к подслучаям того случая, который определяется данным набором.

В качестве примера пакета, использующего разбор случаев, следует взять нормализатор "нормпредел". Перечислим основные вспомогательные процедуры, используемые для организации разбора случаев.

Начнем с процедуры "контрольслучаев(x1 x2 x3)", блокирующей заведомо нецелесообразные попытки разбора случаев. x1 - терм "разборслучаев(или($A_1 \ldots A_n$))"; x2 - список посылок; x3 - список комментариев нормализатора. Блокировка предпринимается, если какое - либо A_i есть константа "истина"; если в x3 содержится элемент "дн"; если список конъюнктивных членов какого-либо A_i содержится в посылках. Кроме того, она предпринимается, если x2 имеет более 3 повторных вхождений посылок.

Если разбор случаев признан целесообразным, то прием прежде всего проверяет наличие комментария "подчинено". Если он есть, то в качестве ответа выдается терм "разборслучаев(...)". Все обращения к пакету рассматриваемого типа из него самого устроены так, что получение промежуточного результата "разборслучаев(...)" вызывает немедленную выдачу этого результата, если есть комментарий "подчинено". Так продолжается до тех пор, пока не будет достигнуто корневое обращение, не имеющее комментария "подчинено". Тогда текущий уровень сканирования в нормализаторе заменяется на 0, и применяется процедура "подслучаи(x1 x2 x3 x4)". В этой процедуре значения переменных x1, x2, x3 воспроизводят значения одноименных переменных нормализатора (преобразуемый терм, посылки и комментарии). x4 - указанное выше выражение "разборслучаев(или($A_1 \ldots A_n$))". Процедура пополняет список подслучаев в комментарии (разборслучаев ...) в соответствии с описанным выше его форматом.

После обращения к процедуре "подслучаи" происходит откат к начальной точке программы пакета. Так как текущий уровень переустановлен на 0, а уровни срабатывания приемов начинаются с 1, то интерпретатор быстро переходит от ветви приемов к концевым фрагментам. Здесь проверяется наличие очередной группы подслучаев, занесенной в комментарий (разборслучаев ...). Из нее извлекается первая альтернатива, по которой переустанавливаются значения x1, x2, x3. Далее текущий уровень устанавливается на 1, и начинается обычная работа нормализатора над выбранным случаем. По завершении преобразований (достижении максимального уровня) интерпретатор снова попадает в ветвь концевых фрагментов.

Если новой серии подслучаев не возникло, то предпринимается обращение к процедуре "очереднойслучай(x1 x2 x3 x4 x5)". Как и выше, x1-x3 суть текущие значения одноименных переменных нормализатора. Процедура регистрирует результат x1 в комментарии (разборслучаев ...). Список подслучаев при этом укорачивается; может произойти завершение рассмотрения текущей серии альтернатив и возвращение к тому пункту предыдущей серии, подслучаи которого анализировались. Если остаются подслучаи для нормализации, то выходной переменной x4 присваивается 1, а переменной x5 - набор $(B_1 \ldots B_6)$, описывающий очередной подслучай. Если все случаи разобраны, то переменной x4 присваивается 0, а переменной x5 - окончательный результат применения нормализатора. Если x3 содержит элемент "подчинено", то разбор случаев отключен - процедура просто передает переменной x5 значение x1, присваивая переменной x4 значение 0.

2.5.9 Процедура "легковидеть"

Если нужно проверить истинность некоторого утверждения F с помощью проверочного оператора, причем из контекста заголовок F и, следовательно, название оператора, не определяются, то можно воспользоваться процедурой "легковидеть(x1 x2 x3 x4)". Она находит заголовок подходящего проверочного оператора и реализует обращение к нему. x1 - проверяемое утверждение; x2 - список посылок; x3 - набор комментариев. Если истинность x1 установлена, то переменной x4 присваивается список использованных посылок. Чтобы определить название оператора, процедура применяет справочник "легковидеть", возникавший ранее при описании компилятора ГЕНОЛОГа.

2.5.10 Процедура "вспомпреобразование"

Пакетный оператор может в особых случаях обращаться к вспомогательной задаче. Чтобы создавать такие задачи, используются процедуры - упрощенные двойники рассмотренных выше процедур "преобразование" и "вспомописание". Начнем с процедуры "вспомпреобразование(x1 x2 x3 x4 x5)", создающей и решающей вспомогательную задачу на преобразование. x1 здесь - преобразуемый терм; x2 - список посылок; x3 - список целей вспомогательной задачи. После решения задачи переменной x4 присваивается результат преобразования; переменной x5 - набор использованных посылок. В списке x3 могут встречаться служебные элементы (уровеньобращения N), (комментарий A), (лимит S), "противоречие", "смпосылка". Первый задает уровень обращения к вспомогательной задаче; второй - передает ей комментарий A; третий - ограничивает трудоемкость; четвертый - разрешает выдачу результата "противоречие", указывающего на противоречивость списка посылок; пятый - понижает до нуля веса посылок вспомогательной задачи. При его отсутствии веса равны 20.

Программа процедуры начинается с рассмотрения буфера, хранящего результаты предшествующих 19 ее применений. Роль буфера играет комментарий (вспомпреобразование A_1 A_2 A_3) к посылкам исходной задачи. Набор A_1 состоит из нулей и троек (x1 x2 x3) входных данных предшествующих обращений; набор A_2 имеет ту же длину и состоит из пар (x4 x5) результатов обращения. A_3 - текущая позиция в A_1, на которую будет заноситься результат очередного обращения. Если в буфере имеется готовый результат, то он выдается. Иначе - устанавливается уровень обращения к вспомогательной задаче; переменной x7 присваивается эта задача, и происходит обращение к ее решению. Отдельно рассматриваются ветви для наличия и отсутствия ограничений на трудоемкость. Если ограничение на трудоемкость превышено, то выход из процедуры по значению "ложь".

В процедуре предусмотрена возможность учета особых случаев, игнорировавшихся приемами вспомогательной задачи x7. Такие приемы оставляют комментарий (контрольодз A) к x7, содержащий список переменных A. После получения ответа находятся все необходимые для сопровождения по о.д.з. отрицания числовых равенств $t_1 \neq t_2$, не усматривающиеся из посылок. В каждом особом случае, определяемом равенством $t_1 = t_2$, преобразование выполняется повторно, причем равенство используется для исключения некоторого параметра. Затем строится итоговое условное выражение, объединяющее все подслучаи. Описанный режим возникает при вычислении интегралов, причем A - переменная интегрирования, а прием, потенциально приводящий к потере особых случаев - процедура перехода к сумме простейших дробей.

После регистрации результата в буфере проверяется, что при отсутствии в x3 элемента "противоречие" ответом вспомогательной задачи не является символ "противоречие", и происходит выдача ответа.

2.5.11 Процедуры "быстрописание" и "Вспомописание"

Чтобы обратиться из пакетного оператора к решению вспомогательной задачи на описание, служит процедура "быстрописание(x1 x2 x3 x4 x5)". Здесь x1 - утверждение, конъюнктивные члены которого образуют список условий задачи; x2 - набор посылок; x3 - список целей. Переменной x4 присваивается ответ задачи; переменной x5 - список использованных посылок. Элементы (уровеньобращения A), (комментарий A), (лимит A) набора x3 используются так же, как в предыдущей процедуре. Вспомогательная задача x8 решается при заблокированном сканировании посылок. Буфер, сохраняющий предыдущие результаты обращений, не используется.

Для вывода теорем создана еще одна вспомогательная процедура, обращающаяся к решению вспомогательных задач на описание - процедура "Вспомописание(x1 x2 x3 x4 x5)". Обращение к ней аналогично процедуре "быстрописание", однако она значительно богаче (например, использует буфер), и по существу является вариантом используемой при сканировании процедуры "вспомописание", ориентированной на другой внешний контекст.

2.5.12 Процедуры пакетных анализаторов

Пакетный анализатор используется для ускоренного получения большой группы следствий посылок некоторой задачи на исследование либо на доказательство и отбора в ней наиболее ценных следствий. Его входные данные суть: x1 - обрабатываемая задача (называется внешней задачей анализатора, причем анализатор преобразует ее копию, называемую внутренней задачей), x2 - максимальный уровень приемов анализатора, используемых при данном обращении к нему, x3 - список целей, передаваемых внутренней задаче. В него может быть включен ограничитель трудоемкости обращения "лимит(N)".

Программа анализатора начинается с обращения к вспомогательной процедуре "анализатор(x1 x2 x3 x4 x5)", где x1,x2,x3 - входные данные анализатора. Эта процедура формирует внутреннюю задачу анализатора x4 и присваивает выходной переменной x5 ограничитель трудоемкости (число шагов работы интерпретатора ЛОСа). Одновременно она используется для организации отлачиком ЛОСа отображения на экране момента обращения к анализатору, а также при создании протокола решения. При создании внутренней задачи анализатора ей передается список неизвестных задачи x1. Если последняя представляет собой задачу на доказательство, то в качестве неизвестных выступают все нечисловые переменные, а также переменные, явно указанные в комментарии (неизвестные ...). Если задача x1 имела цель "контроль", то данная цель передается вспомогательной задаче x4. По умолчанию, лимит x5 полагается равным 2000000. Иначе он определяется термом "лимит(N)", извлекаемым из списка x3. Кроме указанных выше, задаче x4 передаются в качестве целей все отличные от ограничителя трудоемкости элементы списка x3. Так как решение задачи x4 не будет обеспечиваться обращением к процедуре "ответзадачи", то автоматически она текущей задачей не станет. Однако, при работе анализатора ее удобно считать текущей задачей. Поэтому процедура "анализатор" использует обращение к оператору "трассировка(текущаязадача ...)", заносящему ссылку на x4 в регистр текущей задачи. Чтобы при любом выходе из анализатора (в том числе при откате по внешнему "лимиту") происходило восстановление ссылки на старую текущую задачу, здесь же выполняется обращение к оператору "трассировка(откат ...)".

Работа анализатора представляет собой сканирование посылок внутренней задачи. Это сканирование организуется операторами "теквхожд(x1 x2 x3 x4)" либо "корнвхожд(x1 x2 x3 x4)", в зависимости от типа анализатора. Входные данные суть внутренняя задача x3 и текущий уровень сканирования x4. Просматриваются все посылки задачи x3, имеющие вес x4. Выходная переменная x1 перечисляет вхождения логических символов в такие посылки, а выходная переменная x2 - вхождения посылок в список посылок. По окончании просмотра посылки ее вес увеличивается на 1. Оператор "корнвхожд" перечисляет только корневые вхождения x1 посылок задачи x3.

Для перенесения во внешнюю задачу x1 анализатора тех посылок внутренней задачи x2, которые помечены комментарием "внешвывод", используется процедура "внешвывод(x1 x2)". Обычным образом переносятся также посылки, необходимые для сопровождения по о.д.з.

2.6 Разное

2.6.1 Процедуры идентифицирующих операторов

В некоторых случаях идентификация определяемой приемом ситуации требует быстрого просмотра группы объектов, связанных заданными отношениями с уже идентифицированными объектами. Эти отношения, хотя и легко усматриваемые из списка посылок, могут, тем не менее, не содержаться там в явном виде и требовать для своего получения двух-трех простых логических переходов. В принципе, чтобы обеспечить необходимое перечисление, можно было бы создавать пакетные синтезаторы. Однако, при идентификации, в особенности на первых ее этапах, желательно избегать обращений к сколь-нибудь трудоемким процедурам. Поэтому в таких случаях применяются специальные операторы, реализованные непосредственно на ЛОСе. Они называются идентифицирующими операторами. Компилятору сообщаются форматы обращения к ним, для чего служат справочники "усм" и "См". Сами идентифицирующие операторы будут рассматриваться в связи с геометрическими приемами. Здесь ограничимся лишь описанием тех общих средств, которые применяются для ускорения их работы.

Прежде всего, идентифицирующий оператор должен быстро находить все посылки заданного типа, связывающие известный ему объект с другими объектами. Желательно делать это, не прибегая к полному просмотру списка посылок. Поэтому решатель заблаговременно создает ряд вспомогательных древовидных структур данных ("адресных структур"), позволяющих ускорить поиск. Они хранятся в комментариях (списокпосылок ...), (отр ...) и (выражение ...) к посылкам текущей задачи.

Начнем с комментария (списокпосылок A). В нем регистрируются только посылки, заголовки которых отличны от символов "не" и "актив". A - список исходных узлов адресной структуры. Каждый узел (исходный либо внутренний) - пара (B_1, B_2, где B_1 - логический символ либо терм, определяющий признак, по которому разбиваются отнесенные к данному узлу посылки. B_2 - набор пар (C_1, C_2), где C_1 - логический символ либо терм, задающий конкретное значение признака B_1; C_2 - либо список посылок, характеризующихся данным признаком, либо список узлов адресной структуры, в которых сгруппированы посылки, имеющие этот признак.

Используются следующие признаки разбиения посылок:

1. "списокпосылок" - разбиение по заголовкам посылок;
2. "первыйсимвол" либо "второйсимвол" - разбиение по заголовку первого либо второго операнда посылки;
3. "первыйтерм" либо "второйтерм" - разбиение по первому либо второму операнду посылки, который, в отличие от предыдущего случая, берется целиком.

На практике этих признаков и основанного на них двух-трехярусного разбиения оказалось вполне достаточно.

В комментарии (отр A) регистрируются все посылки, имеющие своим заголовком отрицание. Организация данной адресной структуры аналогична организации предыдущей, однако в концевых ее точках посылки регистрируются без отрицаний. Признаки, по которым разбиваются посылки, тоже аналогичны приведенным выше. Однако, при их определении предварительно отбрасывается корневое отрицание. Признак "списокпосылок" при этом переименовывается в "отр", а названия прочих признаков сохраняются.

В комментарии (выражение A) регистрируются все посылки вида "актив(T)". Такие посылки имеют фиктивный характер и являются, по существу, указателями на некоторые выделенные выражения T. Например, в геометрических задачах выделяются рассматриваемые длины отрезков, величины углов, и т.п. Это позволяет инициировать применение приемов с усмотрения выражений соответствующих типов. Организация адресной структуры аналогична двум предыдущим; в концевых ее точках регистрируются выражения T. Используется признак "выражение", значением которого служит заголовок выражения T, а также признак "операнд". Этот признак - многозначный; его значениями служат корневые операнды выражения T. Соответственно, выражение T регистрируется для каждого значения данного признака, и подклассы разбиения, вообще говоря, пересекаются.

Коррекция перечисленных адресных структур выполняется процедурами, преобразующими задачу - "заменавхождения", "вывод", "заменагруппы". Они обращаются для этого к вспомогательной процедуре "смпосылка(x1 x2)". x1 - текущая задача; x2 - вхождение только что измененной или введенной ее посылки. Процедура распадается на три подпроцедуры - "смпосылка1", "смпосылка2" и "смпосылка3". Они корректируют, соответственно, комментарии (списокпосылок ...), (отр ...) и (выражение ...). При удалении посылки тоже необходима коррекция адресных структур; она обеспечивается процедурой "расчистка(x1 x2)", где x2 - вхождение удаляемой посылки. Перечисленные процедуры не инициализируют адресных структур и не создают новых признаков разбиения (кроме признака - корневого заголовка регистрируемого терма); они лишь пополняют группы ссылок для уже имеющихся признаков.

Инициализация адресных структур и признаков разбиения обеспечивается самими идентифицирующими операторами - по мере надобности.

Еще один эффективный способ ускорения идентифицирующих операторов - создание буфера, сохраняющего результаты обращений к ним. Роль такого буфера играет комментарий (Буфер A) к посылкам текущей задачи. Здесь A - набор пар (B C), где B - название идентифицирующего оператора; C - набор пар (входные данные - результат обращения). Если идентифицирующий оператор перечисляющий, то под результатом обращения понимается список всех наборов значений выходных переменных, возникавших при перечислении. Если выходная переменная оператора - единственная, то вместо одноэлементного набора берется сам элемент. Обычно одни и те же идентифицирующие операторы многократно применяются на малом временном промежутке к одним и тем же входным значениям. Поэтому создание буфера равнозначно созданию сетевой структуры данных, устанавливающей прямые связи между рассматриваемыми объектами. При очередном обращении она позволяет сразу найти все объекты, находящиеся в заданных отношениях с заданными объектами, не повторяя ранее проведенных вычислений. Единственный недостаток - необходимость коррекций при каждом изменении списка посылок. Пока используется простейший вид такой коррекции - сброс комментария (Буфер ...). Для занесения новых данных в комментарий (Буфер ...) идентифицирующие операторы используют процедуру "Буфер(x1 x2 x3 x4)". Здесь x1 - текущая задача; x2 - заголовок идентифицирующего оператора; x3 - входные данные; x4 - результат обращения.

2.6.2 Контексты задачи

Приемы часто используют список утверждений, образующий контекст текущей точки задачи. Напомним, что текущее вхождение в задачу определяется тройкой (v_1 v_2 p), где v_1 - вхождение рассматриваемой точки в посылку либо в условие; v_2 - вхождение данной посылки либо условия во внешний список. Таким списком может служить список посылок либо список условий задачи; если рассматривается условие задачи на доказательство либо на преобразование, то этим списком служит сама задача. p - индикатор посылки (0) либо условия (1). Если выделенная точка расположена в посылке, то к ее контексту относятся все прочие посылки; если она выделена в условии, то к контексту относятся все посылки и все прочие условия. Кроме того, добавляются утверждения внешних логических уровней, расположенные в том же терме, что и рассматриваемая точка V. Здесь возможны следующие случаи:

1. V расположено внутри утверждения A_i конъюнкции "и($A_1 \dots A_n$)". Тогда к контексту относятся утверждения $A_1, \dots, A_{i-1}, A_{i+1}, \dots, A_n$.

2. Если V расположено в консеквенте кванторной импликации, то к контексту относятся все антецеденты; если оно расположено в антецеденте, то к контексту относятся все прочие антецеденты.

3. V расположено под описателем "отображение". Тогда к контексту относятся все не содержащие V конъюнктивные члены утверждения, определяющего допустимые значения аргументов.

4. V расположено внутри условного выражения "вариант(P t_1 t_2)". Если оно расположено внутри t_1, то к контексту относятся конъюнктивные члены утверждения P. Если оно расположено внутри t_2, то к контексту относится отрицание

P (в случае неравенств сразу предпринимается замена отрицания на противоположное неравенство).

Для перечисления утверждений контекста служит процедура "обл(x1 x2 x3 x4 x5)". Здесь x1,x2,x3 - координата вхождения V в задачу x4, т.е. x1 $= v_1$; x2 $= v_2$; x3 $= v_3$. Оператор перечисляет все утверждения x5 из контекста вхождения V (в пояснениях к решателю контекст вхождения также называется областью этого вхождения). Если x4 $= 0$, то перечисляются только корневые вхождения утверждений контекста, относящиеся к содержащему V терму. Если x4 $= 1$, то перечисление тоже относится лишь к текущему терму, но перечисляются не вхождения, а сами утверждения.

Если V расположено в условии задачи на описание, то к перечислению добавляются все посылки ее блока анализа, помеченные комментарием "облвхожд".

Если нужно получить список всех утверждений контекста, то используется операторное выражение "облвхожд(x1 x2 x3 x4)". Входные переменные его - те же, что у процедуры "обл". По сравнению с последней процедурой, предпринимаются небольшие модификации списка утверждений контекста, направленные на усиление возможностей быстрого усмотрения его следствий. Результат исключения из контекста всех утверждений заданного списка x5 определяется операторным выражением "Облвхожд(x1 x2 x3 x4 x5)". Еще одна вариация на ту же тему - операторное выражение "конъюнктконтекст(x1 x2 x3 x4 x5)". Входные переменные x1-x4 такие же, как и выше, причем V - вхождение конъюнкции либо кванторной импликации. x5 - набор вхождений в конъюнктивные члены либо в антецеденты и консеквент подтерма V. Значением выражения служит контекст вхождения V, пополненный не содержащими вхождений списка x5 конъюнктивными членами либо антецедентами подтерма V.

Наконец, наиболее простая и быстро определяемая система утверждений контекста - значение K операторного выражения "подобл(x1 x2 x3)". Здесь x1, x2 - вторая и третья компоненты вхождения V в задачу x3. Таким образом, известен лишь терм T, содержащий V, а расположение V внутри этого терма несущественно. В список K заносятся лишь те утверждения контекста, которые находятся вне терма T. Если x3 - задача на описание, имеющая цель "редакция", причем x1 - вхождение условия без неизвестных, то в список K не включаются условия с неизвестными.

2.6.3 Реализация и унификация

Изредка в приемах (главным образом, реализованных на ЛОСе) встречается необходимость подобрать подстановку, переводящую термы заданного списка $t_1, \ldots t_n$ в какие-либо термы другого заданного списка $s_1, \ldots, s_m$. При этом задается список переменных $x_1, \ldots, x_k$, к которым применяется подстановка. Еще реже возникает необходимость подобрать подстановку вместо заданных переменных, отождествляющую термы $t_1, \ldots, t_n$ с термами $s_1, \ldots, s_n$. В первом случае подстановка применяется только к термам $t_1, \ldots, t_n$; эта ситуация называется реализацией данных термов в списке $s_1, \ldots s_m$. Во втором случае подстановка применяется как к термам $t_1, \ldots, t_n$, так и к термам $s_1, \ldots, s_n$; эта ситуация называется унификацией термов. Существенным является то обстоятельство, что при реализации и унификации разрешаются несложные дополнительные преобразования термов; например, перестановка операндов коммутативных операций и отношений.

Редкое использование в решателе общих процедур унификации и реализации объясняется тем, что обычно приемы основаны на конкретных теоремах, и реализацию выполняет создаваемая компилятором процедура идентификации. Она-то и определяет подстановку вместо теоремных переменных, переводящую термы теоремы в термы задачи. Процедура унификации могла бы быть полезной при выводе следствий из двух кванторных импликаций, однако режим вывода таких следствий оказался невостребованным. Обычно кванторные импликации используются в режиме вывода элементарных утверждений. Впрочем, процедура унификации применяется в некоторых приемах вывода теорем. В старых версиях вывода теорем процедура унификации применялась исключительно часто, чем и объясняется то, что она оказалась существенно более развитой, чем процедура реализации. Иногда она используется вместо процедуры реализации, как более сильная. Тогда переменные, к которым применяется подстановка, входят лишь в реализуемые термы $t_1, \ldots, t_n$.

Обращение к процедуре реализации имеет вид "реализация(x1 x2 x3 x4)". Здесь x1 - набор элементов, уточняющих задачу реализации; x2 - набор переменных, по которым выполняется подстановка. Если попытка реализации удалась, то переменной x3 присваивается набор термов, подставляемых вместо переменных x2; переменной x4 - список неиспользованных реализующих термов. Оператор работает в режиме перечисления. В наборе x1 используются элементы двух типов:

1. (реализация A_1 A_2). Здесь A_1 - набор пар термов $(s\ t)$, перечисляющий такие реализуемые термы t, для которых заранее указан соответствующий реализующий терм s. Набор A_2 относится к разбиению реализующих и реализуемых термов на группы; он состоит из четверок $(S\ T\ f\ p)$. T - набор реализуемых термов; S - набор реализующих термов; f - заголовок внешних сравниваемых операций, операндами которых служили наборы S, T (может стоять фиктивное значение); p - указатель полного (0) либо неполного (1) использования элементов набора S. При реализации элементов T соответствующие элементы из S берутся в произвольном порядке, причем каждый используется не более чем однократно.

2. (заголовок A). A - набор, длина которого равна длине набора переменных x2. Он состоит из наборов допустимых ассоциативных заголовков подставляемых вместо данных переменных термов. Ноль в наборе A означает, что допустим произвольный заголовок.

Перейдем к программе процедуры. Выйти на нее можно через раздел оглавления программ "Приемы решателя" - "Общие процедуры, используемые в приемах" - "Процедура РЕАЛИЗАЦИЯ".

Переменной x5 присваивается элемент (реализация ...) набора x1. Предполагается, что такой набор в x1 единственный. Переменной x6 присваивается бланк для набора подставляемых термов. Вводится накопитель x7 остаточных реализующих термов. Процедура реализации состоит из группы продукций (приемов), выполняющих преобразования установки на реализацию x5 и накопителей результата x6, x7. Срабатывание этих продукций выполняется на различных "уровнях сканирования" установки x5. Роль указателя текущего уровня играет переменная x8, инициализируемая нулем.

После контрольной точки "прием(14)" реализуется цикл создания элемента (заголовок A), регистрируемого в списке x1. При первом обращении этот элемент обычно отсутствует, однако бывает полезен при рекурсивных обращениях. Он ускоряет отсечение непригодных вариантов. Переменной x9 присваивается бланк набора A. Для каждой переменной x списка x2 сначала рассматриваются пары (s,t) набора A_1 установки (реализация A_1 A_2). Находятся всевозможные вхождения в реализуемый терм t операций $f(\ldots x \ldots)$, и составляется список C всех ассоциативных логических символов, являющихся заголовками операндов q_i подтермов $f(q_1 \ldots q_m)$ терма s. Если f ассоциативно, то оно тоже заносится в C. Аналогичные списки C составляются для четверок набора A_2, где берутся всевозможные пары термов t, s групп T,S. Все эти списки C пересекаются, и результат заносится на позицию списка x9, соответствующую переменной x.

Далее начинается цикл преобразований установки на реализацию. Перечислим применяемые здесь приемы. Как и выше, предполагаем, что текущая установка на реализацию есть (реализация A_1 A_2). Попытки применения приемов осуществляются в том порядке, в каком эти приемы перечисляются. После каждого срабатывания приема очередной просмотр начинается с начала списка приемов.

1. Пусть в A_1 имеется такая пара (s, t), у которой терм t не содержит переменных списка x2. Если результаты простейшей стандартизации термов s, t совпадают (упорядочение операндов коммутативных операций; устранение вложенных ассоциативных операций, и т.п.), то данная пара исключается из A_1. Иначе - выход из процедуры по значению "ложь". Этот и следующий приемы расположены в программе после контрольной точки "прием(16)".

2. Пусть в A_2 имеются четверка (S T f p) и терм t из T, не содержащий переменных списка x2. Если ни t, ни результат стандартизации t не содержатся в S, то выход из процедуры по значению "ложь". Иначе - t удаляется из T и из S. Если S становится пустым, в то время как T непусто, то выход по значению "ложь". Если T становится пустым, в то время как S непусто и $p = 1$, то S добавляется к накопителю x7. Во всяком случае, при пустом T рассматриваемая четверка удаляется из A_2.

3. Если в A_1 имеется пара (s, $g(t)$), у которой символ g удовлетворяет тождеству $g(g(x)) = x$, то рассматриваются два подслучая: t имеет вид $g(r)$ либо t - переменная списка x2.В первом подслучае предпринимается замена $g(f(r))$ на r; во втором - замена пары на ($g(s)$, t). Прием расположен после контрольной точки "прием(12)".

4. В A_1 рассматривается пара ($f(s_1 \ldots s_n)$, $g(t_1 \ldots t_m)$). Если символы f, g различны, то выход по значению "ложь". Это же происходит, если $m \neq n$, причем неверно, что символ f является ассоциативно-коммутативным. Иначе, при некоммутативном f, пара заменяется в наборе A_1 на группу пар (s_1, t_1), ..., (s_n, t_n). При коммутативном f пара исключается из A_1, а в A_2 заносится четверка ($(s_1, \ldots, s_n)$, $(t_1, \ldots, t_m)$, f, 0). Отдельно рассмотрены два частных случая символа f - "угол"и "точкалуча". Здесь имеется специальная симметрия, дающая два альтернативных случая реализации. Для обработки их используется рекурсивное обращение к процедуре. Прием расположен после контрольной точки "прием(5)".

5. В A_1 рассматривается пара (s x), где x - переменная списка x2. На соответствующую переменной x позицию накопителя результата x6 заносится терм s. Все вхождения переменной x в реализуемые термы для наборов A_1, A_2 заменяются на s. Прием расположен после контрольной точки "прием(4)".

6. Если списки A_1, A_2 пусты, то выдается результат - содержимое накопителей x6, x7.

7. В A_2 рассматривается четверка $(S\ T\ f\ p)$. Если список S короче списка T, то выход по значению "ложь". Если список T состоит из единственного терма t, причем либо t - переменная из x2, либо S одноэлементно, либо $p = 0$, то четверка удаляется из A_2, а в A_1 заносится пара $(f(s_1, \ldots, s_n), t)$. Здесь $S = \{s_1, \ldots, s_n\}$. Прием расположен после контрольной точки "прием(7)".

8. Если в A_2 есть четверка, некоторый реализуемый терм которой имеет вид $g(g(r))$, причем операция g удовлетворяет тождеству $g(g(x)) = x$, то данный терм заменяется на r. Прием расположен после контрольной точки "прием(13)".

9. Предпринимается просмотр четверок $(S\ T\ f\ p)$ набора A_2, чтобы отобрать такой реализуемый терм $t \in T$, который дает наименьшую трудоемкость последующего разбора случаев. Цикл просмотра начинается с контрольной точки "прием(6)". Переменная x10 играет роль накопителя отбираемой для разбора случаев четверки; x11 - накопитель отбираемого терма t; x12 - пара (N, M) численных характеристик текущего варианта. Здесь N - число термов списка S, имеющих одинаковый заголовок с t; M - для случая ассоциативного заголовка t есть число корневых операндов реализующего терма, ограничиваемых с помощью элемента (заголовок $\ldots$). Из просмотра исключаются случаи термов t вида $g(y)$, где $g(g(x))$ тождественно равно x, а y - переменная списка x2. Если при просмотре оказывается, что S не имеет терма с тем же заголовком, что и терм t, то выход по значению "ложь". Если оказывается, что такой терм s в списке S единственный, то в набор A_1 заносится пара $(s\ t)$, а из списков S, T термы s, t исключаются. Если список T остается пустым, то в зависимости от пустоты S и значения p либо выдается значение "ложь", либо пополняется накопитель x7. Отбор терма t выполняется по принципу минимизации значения N; при равенстве таких значений предпочтение отдается не коммутативному заголовку терма t; в прочих случаях - при ассоциативном заголовке максимизируется M.

   По окончании просмотра - переход через "иначе 1" к фрагменту, содержащему контрольную точку "прием(15)". Здесь временно откладываются отобранные выше четверка x10 и терм x11; вместо этого отбираются такие четверка $(S\ T\ f\ p)$ и реализуемый терм t из T, представляющий собой переменную x списка x2 либо имеющий вид $g(x)$ $(g(g(x)) = x)$, что элемент (заголовок $\ldots$) накладывает непустые ограничения на заголовок соответствующего переменной x терма, и при этом число элементов списка S - наименьшее. Эти четверка и терм присваиваются, соответственно, переменным x14 и x15. Затем перебираются всевозможные варианты сопоставления элемента списка S терму t, и для каждого из них реализуется рекурсивное обращение к процедуре. По мере обработки вариантов, выдаются результаты реализации.

   Если указанный выше отбор оказался невозможным, то откат к переходу через "ветвь 1". Возникает фрагмент, содержащий контрольную точку "прием(10)". Здесь возвращаемся к отобранным ранее четверке x10 $= (S\ T\ f\ p)$ и терму x11 $= t$. Если x10 отлично от 0, то последовательно просматриваются варианты реализации, при которых терму t сопоставляется терм s списка S, имеющий с ним общий заголовок. Для каждого из этих вариантов выполняется рекурсивное обращение к процедуре, с выдачей текущих результатов реализации.

> Наконец, если x10 равнялось 0, то предпринимается последняя попытка реализации (контрольная точка "прием(11)"). Она состоит в отборе такой переменной x списка x2, которая является реализуемым термом сразу для нескольких четверок, причем во всех этих четверках используется одна и та же внешняя ассоциативно-коммутативная операция f. Для сопоставления переменной x находится пересечение всех групп реализующих термов указанных четверок, к которому применяется операция f.

Перейдем к процедуре "унификация(x1 x2 x3)". В обращении к ней указываются набор установок на унификацию x1 и список переменных x2, по которым выполняется унификация. x3 перечисляет наборы термов, подставляемых при унификации вместо переменных x2. Как уже говорилось выше, при унификации допускаются различные простые преобразования термов. Это отличает ее от используемой в математической логике "однозначной" унификации и приводит к необходимости режима перечисления результатов.

В наборе x1 используются следующие типы элементов (многие из них вводятся самой процедурой и нужны при рекурсивных обращениях):

1. (унификация t_1 t_2 K). Требуется унифицировать термы t_1 и t_2 (отождествить их с помощью результирующей подстановки, быть может, с точностью до некоторых простых тождественных или эквивалентных преобразований). K - набор вспомогательных пометок, вводимых процедурой;
2. (переменные A). A есть список всех переменных термов, вовлеченных в унификацию. Элемент нужен при рекурсивных обращениях, когда часть внешних переменных может исчезнуть из текущих унифицируемых термов. Так как процедура унификации может вводить новые переменные, ей требуется информация об уже использованных ранее переменных;
3. (терм A_1 A_2). A_1, A_2 суть исходные теоремы, из которых извлечены унифицируемые термы. Необходимо по той же причине, что и предыдущее - для определения ранее использованных переменных;
4. (результподст A). A есть переданный при рекурсивном обращении набор термов, уже зафиксированных для подстановки вместо переменных списка x2;
5. (единица A). При унификации допускается подстановка единиц вместо переменных списка A. При подстановке единицы внешняя двуместная операция отбрасывается;
6. "приведение". Блокировка стандартизации промежуточных термов оператором "приведение";
7. (независит A_1 A_2). Терм, унифицированный с переменной A_2, не должен зависеть от переменной, унифицированной с переменной A_1. Элемент возникает при унификации описателей и кванторов;
8. (значение A_1 A_2). Терм, унифицированный с переменной A_2, может зависеть от переменной, унифицированной с переменной A_1.
9. "новаяпеременная". Блокируется введение новых переменных;

10. (списокпеременных A). A есть список новых переменных, введенных при унификации;
11. "заменазнака". При унификации разрешается использование преобразований, определяемых справочником "заменазнака";
12. "отрицание". Блокировка преобразований, определяемых справочником "отрицание".

Процедура унификации создавалась для обслуживания первых версий циклов вывода в базе теорем. Ее усиление приводило к слишком большому числу рассматриваемых вариантов и длительному перебору; ослабление - к потере важных следствий. В итоге определился некоторый промежуточный вариант. В целом, состав вошедших в этот вариант преобразований имеет эвристический характер; некоторые из них ориентированы на весьма частные случаи.

Еще раз напомним, что в решателе процедура унификации практически не используется. Данный режим сложился в процессе "естественного отбора в первоначальных версиях процедуру предполагалось применять в составе приемов общелогического характера. Однако, первые же шаги оптимизации решателя потребовали усиленной блокировки таких приемов, а впоследствии и полного их исключения. После этого унификация долгое время оставалась центральным блоком аппарата вывода в базе теорем. Однако, в последних версиях она и отсюда оказалась практически исключенной. Это не означает полного отказа от применения унификации - она составляет важный резерв для тех этажей логической системы, которые должны будут обеспечивать пополнение базы приемов вывода теорем.

Переходим к рассмотрению программы, которую нетрудно найти в том же разделе оглавления программ, в котором размещена процедура "реализация".

Прежде всего, проверяется наличие в списке x1 элемента (терм ...), и если его не было, то такой элемент создается. Точно так же, при необходимости создается элемент (переменные ...). Переменной x4 присваивается заготовка набора подставляемых термов. Она инициализируется набором переменных x2, преобразованных к формату термов. Если в x1 есть элемент (результподст ...), содержащий заготовку результата из внешней процедуры, то эта заготовка переприсваивается переменной x4. Далее начинается продукционная часть процедуры, распадающаяся на отдельные приемы. Попытки применения приемов упорядочены по уровням; значение текущего уровня присвоено переменной x5. После контрольной точки "прием(2)" размещен цикл просмотра установок (унификация t_1 t_2 K), имеющихся в списке x1. Текущая такая установка присвоена переменной x7; терм t_1 присвоен переменной x8, а терм t_2 - переменной x9. В течение цикла просмотра значение уровня x5 фиксировано. По окончании цикла проверяется отсутствие установок (унификация ...). Если их нет, то выдается результат x4; иначе - значение уровня увеличивается на 1, и откат к следующему циклу. По достижении уровня 10 - выход из процедуры по значению "ложь".

Перечислим используемые процедурой приемы, упорядочив их по возрастанию уровня. Будем указывать уровни лишь для тех приемов, где происходит их увеличение на 1. Первые приемы имеют уровень 0.

1. Если $t_1 = t_2$, то установка x7 удаляется. Прием расположен после контрольной точки "прием(3)".

2. После контрольной точки "прием(9)" рассматриваются подслучаи, в которых термы t_1, t_2 имеют различные заголовки и отличны от переменных списка x2. Перечислим эти подслучаи.

   (a) Один из термов t_1, t_2 имеет вид $g(x)$, где x - переменная списка x2, а g - операция, удовлетворяющая тождеству $g(g(y)) = y$. Другой терм указанной пары обозначим t. В x1 нет элемента "отрицание". Тогда в установке на унификацию рассматриваемая пара термов заменяется на термы x, $g(t)$. Прием расположен после контрольной точки "прием(46)";

   (b) Один из термов t_1, t_2 имеет вид "степень(x n)", где x - переменная списка x2; n - натуральная константа. Другой терм t - произведение двух сомножителей. В x1 нет элемента "новаяпеременная". Выбираются две новые переменные y, z, и предпринимается рекурсивное обращение к унификации, в котором установка x7 заменена на две установки - унификацию терма "умножение(степень(y n)степень(z n))" с термом t и терма x с термом "умножение(y z)". После исчерпания результатов данного рекурсивного обращения - выход по "ложь". Прием расположен после контрольной точки "прием(15)";

   (c) Один из термов t_1, t_2 имеет вид $f(p\ ,\ x)$, где x - переменная списка x2, причем операция f имеет единицу a для своего операнда x. Переменная x упомянута в элементе (единица ...) списка x1. Дргуой терм рассматриваемой пары - t. Предпринимается рекурсивное обращение к унификации, в котором установка x7 заменена на две установки - унификацию терма p с t, а терма x с a. По исчерпании результатов этого рекурсивного обращения - переход к рассмотрению прочих способов унификации. Прием расположен после контрольной точки "прием(5)";

   (d) Один из термов t_1, t_2 имеет вид $f(g(r_1 \dots r_n))$; другой - вид $g(v_1 \dots v_m)$. В x1 отсутствует элемент "приведение". Согласно справочнику "заменазнака", возможно перенесение операции f на операнд r_i. Тогда предпринимается рекурсивное обращение к унификации пары термов $g(r_1 \dots f(r_i) \dots r_n)$, $g(v_1 \dots v_m)$. В случае коммутативного g перебираются всевозможные операнды r_i. Прием расположен после контрольной точки "прием(10)";

   (e) В остальных подслучаях - выход из процедуры по значению "ложь".

3. Терм t_1 представляет собой переменную списка x2, причем терм t_2 либо не является такой переменной, либо в x1 отсутствует элемент (значение ...), последний разряд которого равен t_2. Последнее требование определяет приоритеты при унификации двух переменных: переменные, упомянутые в элементах (значение ...), исключаются в первую очередь.

   Ситуация рассматривается после контрольной точки "прием(7)"; уровень приема равен 1.

   Переменной x10 присваивается переменная - заголовок терма t_1. Если она входит в терм t_2, то выход по значению "ложь". Иначе - после контрольной точки "прием(11)" к каждому терму накопителя результата x4 применяется подстановка терма t_2 вместо x10. Результаты применения подстановки упрощаются с помощью процедуры "приведение". Эта процедура выполняет ряд элементарных преобразований - устранение "двойных отрицаний"; "единичных операндов"; вложенных ассоциативно-коммутативных одинаковых операций, и т.п.

После изменения накопителя x4 проверяется выполнение условий, накладываемых элементами (независит ...) набора x1. При нарушении условий - выход по "ложь". Текущая установка на унификацию x6 удаляется из списка x1; уровень x5 устанавливается на 0; в оставшихся установках на унификацию выполняются подстановка t_2 вместо x10 и упрощение результатов подстановки. Затем - откат к началу цикла просмотра приемов.

4. Терм t_2 представляет собой переменную списка x2. Прием начинается после контрольной точки "прием(13)". С точностью до порядка термов, он аналогичен предыдущему приему.

5. Заголовком терма t_1 служит некоммутативный символ. Случай рассматривается после контрольной точки "прием(19)"; уровень приема равен 2. Заголовок терма t_1 присваивается переменной x10. Проверяется, что термы t_1, t_2 неоднобуквенные. Заголовки их, согласно ранее применявшимся приемам, должны быть равны. Определяются наборы $(v_1, \ldots, v_n)$ и $(w_1, \ldots, w_n)$ корневых операндов данных термов, присваиваемые переменным x11 и x12. Если длины этих наборов различны, то выход по значению "ложь".

   Если $n = 1$, причем v_1 и w_1 имеют вид "набор($P_1 \ldots P_k$)" и "набор($Q_1 \ldots Q_s$)", а одноместная операция x10 не зависит от порядка элементов набора (например, x10 = "перечень"), то вместо v_1, w_1 в списках x11, x12 помещаются термы "и($P_1 \ldots P_k$)", "и($Q_1 \ldots Q_s$)". Здесь "и" - произвольно взятая коммутативная операция, позволяющая в дальнейшем игнорировать порядок операндов.

   Текущая установка унификации x6 заменяется на группу установок унификации v_i с w_i; $i = 1, \ldots, n$. Уровень x5 устанавливается на 0. Перед тем, как выполнить откат к началу цикла просмотра приемов, рассматриваются случаи, когда x10 - символ "отображение" либо "класс". Перебираются всевозможные пары переменных x, y, где x входит в связывающую приставку одного из описателей t_1, t_2, а y - свободная переменная выражения либо утверждения под описателем. Если в x1 отсутствует элемент (значение x y), разрешающий зависимость y от x, то вводится элемент (независит x y), блокирующий такую зависимость.

6. Заголовок x10 - коммутативный символ. Случай рассматривается после контрольной точки "прием(24)". Здесь сосредоточена бóльшая часть приемов унификации. Переменным x11 и x12 присваиваются списки $v_1, \ldots, v_n$, $w_1, \ldots, w_m$ корневых операндов термов t_1, t_2. Переменной x13 присваивается пересечение списков x11, x12.

   Если x13 непусто, то находятся результаты x14, x15 исключения его элементов из x11, x12. Проверяется, что либо списки x14, x15 оба непусты, либо оба они пусты. Если это не так, но операция x10 имеет единицу e, выполняется замена пустого списка на одноэлементный набор (e). Иначе - выход по значению "ложь". Если оба списка оказались пусты, то текущая установка x6 удаляется, уровень x5 устанавливается на 0, и откат. Если оба списка непусты, то термы t_1, t_2 заменяются в текущей установке на результаты применения операции x10 к их элементам, уровень x5 устанавливается на 0, и откат.

   Если пересечение x13 списков x11, x12 пустое, а операция x10 ассоциативна, то после контрольной точки "прием(37)" переменной x14 присваивается один из списков x11, x12, а переменной x15 - другой. Затем рассматриваются следующие подслучаи:

(a) Ни один из термов списка x14 не является переменной унификации. Если список x14 короче списка x15, то выход по значению "ложь".

(b) Ни один из термов списка x14 не является переменной унификации. Число термов списка x15 с некоторым отличным от переменной унификации заголовком a больше числа таких термов списка x14. Если x1 не имеет элементов "заменазнака" и (единица S), где S пересекается с параметрами списка x14, то выход по "ложь".

(c) Ни один из термов списка x14 не является переменной унификации. В x1 входит элемент "заменазнака". Составляется список x16 всех одноместных операций, которые согласно справочнику "заменазнака" можно переносить на операнды встречающихся в x14 и x15 термов $f(t_1 \dots t_n)$; $n > 1$. Проверяется отсутствие в x14 терма вида $g(x)$, где x - переменная унификации, а g - символ из x16. Пусть оказывается, что число возникающих после отбрасывания внешней операции из x16 термов списка x15 с некоторым отличным от переменной унификации заголовком a больше числа таких термов для списка x14. Если x1 не имеет элемента (единица S), где S пересекается с параметрами списка x14, то выход по "ложь".

(d) Ни один из термов списка x14 не является переменной унификации. Выбирается такой отличный от переменной унификации терм r списка x15, для которого число N имеющих тот же заголовок термов списка x14 - наименьшее возможное. Если $N = 1$, то для единственного терма s списка x14, обладающего таким заголовком, создается новая установка на унификацию его с r. Из текущей установки на унификацию оба терма исключаются (если они были в списках x14, x15 единственные, то текущая установка удаляется), и откат к началу цикла просмотра приемов. Если $1 < N < 6$, то процесс унификации разветвляется - последовательно выполняются рекурсивные обращения, соответствующие унификации r с различными термами s списка x14, имеющими тот же заголовок. Перечисленные действия выполняются на различных уровнях; величина уровня равна $N+5$. Прием расположен после контрольной точки "прием(33)".

(e) Пусть длины списков x14 и x15 одинаковы, причем некоторый терм r списка x14 не является переменной унификации либо ее "отрицанием". Если список x15 не имеет терма с таким же заголовком или терма, преобразуемого к нему с помощью справочника "заменазнака", но в x15 имеется единственный терм - переменная унификации x, то выполняется сопоставление переменной x терма r. Затем - откат. Уровень срабатывания приема равен 6, а расположен он после контрольной точки "прием(36)". Имеется версия данного приема, в которой заблокировано использование справочника "заменазнака"; ее уровень равен 7.

7. Для ассоциативной операции x10, далее, реализуются следующие приемы, в которых уже не используются введенные выше списки x14 и x15:

   (a) Последовательно просматриваются термы r списка x11, заголовок которых отличен от переменной унификации и для которых в комментариях к текущей установке на унификацию отсутствует пометка (стоп r). Из них выбирается терм, имеющий наименьшее число N термов s списка x12 с тем же заголовком, и последовательно разбираются случаи унификации

r с такими s. Допускаются значения N от 1 до 4; при $N = 1$ уровень срабатывания равен 7, при $N = 2$ - 8, и при $N = 3, 4$ - 9. По окончании рассмотрения указанных подслучаев вводится пометка (стоп r), и откат.

(b) Список x11 состоит из переменной унификации x и некоторого другого терма r. В списке x12 имеется переменная унификации y. Либо список x12 двухэлементный и длина терма r меньше 9, либо длина этого терма меньше 4. Предпринимается попытка унификации y с r, а x - с остатком термов списка x12. Прием расположен после контрольной точки "прием(14)". Уровень его и всех последующих приемов данной серии равен 10.

(c) В списке x11 имеется переменная унификации x. Список x12 - двухэлементный, состоит из переменной унификации y и некоторого однобуквенного терма s. Предпринимается попытка унификации x с s, а y - с остатком термов списка x11.

(d) Заголовок унифицируемых термов f (т.е. x10) не есть символ "и". В списке x11 выделяется переменная унификации x; в списке x12 - переменная унификации y. Пусть A, B - остатки списков x11, x12. Либо в A, B отсутствуют переменные унификации, либо каждый из этих списков одноэлементен. В B нет термов, содержащих x; в A нет термов, содержащих y. Вводится новая переменная z и предпринимается попытка унифицировать x с $f(z, B)$, а y - с $f(z, A)$. Прием расположен после контрольной точки "прием(40)".

(e) Число переменных унификации в одном из списков x11, x12 не менее двух. Обозначим этот список x14, а другой список - x15. Рассматриваются случаи, когда списки x14, x15 имеют не более трех элементов. В этих случаях перебираются сопоставления заданной переменной унификации первого списка единичным термам второго списка. Прием расположен после контрольной точки "прием(41)". После контрольной точки "прием(16)" расположен вариант данного приема, относящийся к двухэлементным спискам x14, x15 и использующий справочник "отрицание".

8. Наконец, после контрольной точки "прием(45)" рассматривается случай коммутативного, но неассоциативного заголовка x10 унифицируемых термов. Предполагается, что оба эти терма имеют ровно два корневых операнда, и рассматриваются две возможности их сопоставления друг другу. Уровень приема равен 2.

Имеется усиленная версия оператора "реализация", использующая обращение к процедуре унификации. Обращение к ней имеет вид "Реализация(x1 x2 x3 x4)". Здесь x1, x2 - термы; x3 - набор переменных. Если терм x2 получается из x1 подстановкой вместо переменных списка x3 некоторых термов и применением простейших преобразований, допускаемых процедурой унификации, то x4: = список подставляемых термов. Перед обращением к процедуре "унификация" здесь происходит переобозначение в x1 переменных списка x3 на новые переменные x5. Затем унификация проводится по списку x5.

В заключение раздела рассмотрим процедуру "подборпосылок(x1 x2 x3 x4 x5 x6 x7 x8)", применяемую в некоторых приемах и обращающуюся к процедуре "реализация". x1 - задача; x2 - набор утверждений; x3 - набор переменных, не входящих в посылки задачи x1. Процедура обеспечивает подбор таких термов, подстановка которых в утверждения x2 вместо переменных x3 дает (с точностью до простых преобразований) некоторые посылки задачи x1. Если $x4 = 0$, то указанным образом "реализуются" все утверждения списка x2; если $x4 = 1$, то реализуются все утверждения, кроме какого-либо одного (произвольного оставшегося). x5 - список посылок задачи x1, которые запрещается использовать. При успехе переменной x6 присваивается найденный набор термов. Переменной x7 присваивается результат применения реализующей подстановки к отрицанию оставшегося нереализованным утверждения списка x2 (при $x4 = 0$ этой переменной присваивается 0). Переменной x8 присваивается набор использованных посылок (реализующих либо использованных для обоснования корректности преобразований).

Перейдем к программе процедуры. Прежде всего, в списке x2 выбирается наиболее однозначно реализуемое утверждение. С этой целью составляется список x9, перечисляющий без повторений пары (заголовок реализуемого утверждения с отброшенным внешним отрицанием - указатель наличия отрицания). Для каждой пары списка x9 находится число не вошедших в список x5 посылок задачи x1 с данными заголовком и указателем отрицания, и находится пара x13 с наименьшим таким числом. Выбирается какое-то утверждение x14 списка x2, соответствующее паре x13. Переменной x15 присваиваются все параметры этого утверждения, входящие в список x3.

Последовательно просматриваются не входящие в список x5 посылки x16 задачи x1, соответствущие паре x13. Для текущей такой посылки выполняется обращение к процедуре "реализация", где роль реализуемого терма играет x14, роль реализующего - x16, а переменные реализации суть x15. В случае успеха определяется список x20 результатов применения реализующей подстановки к отличным от x14 термам набора x2. Если список x20 пуст, либо $x4 = 1$ и длина списка равна 1, то выдается результат. Иначе - рекурсивное обращение к процедуре "подборпосылок" для остатка переменных x3, не вошедших в список x15, где роль реализуемых играют утверждения списка x20. При этом посылка x16 объявляется запрещенной.

Если просмотр посылок закончился безрезультатно, причем $x4 = 1$, то принимается решение о реализации всех отличных от x14 утверждений списка x2. Для этого выполняется рекурсивное обращение к процедуре "подборпосылок", где значение входной переменной x4 полагается равным 0. Если и эта попытка завершается безрезультатно, реализуется выход по значению "ложь".

2.6.4 Переключение внимания

Для уменьшения весов посылок или условий задачи компилятор ГЕНОЛОГа обычно создает развернутые кванторные импликации, не прибегая к специальным процедурам переключения внимания. В редких случаях используется процедура "внимание(x1 x2)". Здесь x1 - задача; x2 - терм либо логический символ. Процедура находит все условия и посылки задачи x1, содежащие терм x2, и заменяет их веса на 0. Если x2 - логический символ "условие", причем x1 - задача на преобразование либо на доказательство, то заменяется на 0 вес условия этой задачи. Программа процедуры несложна.

Глава 3

Общелогические приемы, реализованные на ГЕНОЛО́Ге

Общелогические приемы, реализованные на ГЕНОЛО́Ге, лишь дополняют главную часть таких приемов, реализованную на ЛОСе. Как правило, они имеют очень специальный характер. Исключение составляют несколько больших пакетов - "нормили", "нормсуществует", "нормдлялюбого", "нормвариант", и др. Они часто используются решателем в тех ситуациях, когда сканирование задачи по каким-либо причинам невозможно. Например, нормализатор "нормили" обеспечивает завершающее редактирование ответа произвольной задачи на описание, имеющего вид дизъюнкции. Это предопределяет наличие в нем большого числа приемов, имеющих, по существу, уже далеко не общелогический характер. Таким образом, применение здесь ГЕНОЛОГа оказывается вполне оправданным.

Смысл записанного на ГЕНОЛОГе общелогического приема часто определяется не столько его теоремой, сколько дополнениями к ней. Иногда теорема вообще нужна лишь как техническая формальность, позволяющая организовать обращения к необходимым вспомогательным процедурам. Это существенным образом отличается от использования ГЕНОЛОГа в конкретных разделах, где "псевдотеоремы" приемов возникают редко. Там теорема приема, как правило, очень близка к обычной теореме и несет в себе главную часть информации о приеме.

Общелогические приемы накапливались в процессе обучения решателя достаточно бессистемно. Из общих соображений, в отрыве от потребностей обработки текущего обучающего материала, они не создавались. Это же, впрочем, относится и к остальным разделам. Причина проста: приемы, созданные из общих соображений, чаще всего оказываются либо не востребованы решателем, либо портят уже отлаженные процессы решения задач.

В дальнейшем для поиска приема будем пользоваться не оглавлением программ, а оглавлением приемов (клавиша "г" из главного меню). В ближайшее время нас будет интересовать подраздел данного оглавления "Логические приемы" - "Общие приемы".

3.1 Конъюнкция

В этом разделе представлены несколько совсем простых приемов.

Лексикографическое упорядочение операндов

Все приемы такого рода имеют теорему "коммутативно(A)", где A - рассматриваемое коммутативное отношение, операция либо логическая связка. Заголовок приема - "лексупорядочение". Единственный фильтр, которым должен обладать прием - определитель уровней срабатывания "уровень($u_1 u_2$)". Обычно это фильтр "уровень(0 3)". Здесь u_1 - уровень, на котором прием срабатывает в посылках; u_2 - уровень, на котором он срабатывает в условиях. Для простых приемов ГЕНОЛОГа можно рекомендовать в качестве упражнения просмотр соответствующих им программ. Переход к программе - по клавише "Home"; обратный переход - "End".

Устранение вложенных конъюнкций

Приемы для устранения вложенных коммутативно-ассоциативных операций A имеют такую же теорему, как и в предыдущем случае - "коммутативно(A)". Заголовок приема - символ "спускоперандов". Прием может иметь произвольные фильтры, хотя чаще всего - единственный фильтр "уровень(0)". Заметим, что в данном случае теорема приема должна была бы содержать также требование ассоциативности символа A. Однако, для компилятора эта теорема вообще не нужна, так что ее роль - по существу, роль заглушки.

Отрицание конъюнкции

Первый в данном разделе прием, для которого теорема несет хоть какую-то используемую компилятором информацию. Он преобразует отрицание конъюнкции в дизъюнкцию отрицаний, если хотя бы один из конъюнктивных членов имеет своим заголовком отрицание: $\forall_{ab}(\neg(\neg a \ \& \ b) \leftrightarrow (a \vee \neg b))$. Дополнительно требуется, чтобы рассматриваемое отрицание конъюнкции представляло собой содержащее неизвестные условие задачи на описание, было достаточно длинным (200), и не имел места этап редактирования найденного ответа. Заметим, что хотя в теореме приема конъюнкция имеет всего два операнда, но идентифицироваться она будет с конъюнкцией произвольной длины: все операнды, кроме идентифицированного с $\neg a$, будут отнесены к b. Уровень срабатывания приема равен 1. Преобразованная дизъюнкция снабжается комментарием "разборслучаев", форсирующим переход к рассмотрению альтернатив. Заметим, что ранее уже встречался прием, преобразующий отрицание конъюнкции в дизъюнкцию отрицаний. Однако, он срабатывал на уровне 2. Данный прием - версия с ускоренным срабатыванием. Видимо, в процессе обучения решателя возникала ситуация, когда требовался ускоренный разбор случаев. В качестве упражнения рекомендуется найти те задачи задачника, в которых прием срабатывает, и повторно проанализировать необходимость его использования. Такого рода приемы лишь обозначают возникавшие трудности, и их можно рассматривать как исходный материал для последующей оптимизации решателя.

Справочники

Приводятся приемы справочников "ассоциативно", "коммутативно", "монотонно". Первые два справочника используются очень часто, третий - вообще практически не используется. Вручную вводить такие приемы нет необходимости. Достаточно ввести лишь их теорему - "ассоциативно(A)" либо "коммутативно(A)", после чего воспользоваться генератором приемов. Нажимается клавиша "А" (кир.); в оглавлении типов приемов выбирается первый пункт ("Справочники и сопровождающие их простейшие приемы"); нажимается "курсор вправо". Генератор приемов создает один или несколько приемов, основанных на введенной теореме, и прорисовывает на экране первый из них. Чтобы выбрать его, нажимается F3; чтобы перейти к следующему приему - "ш". Эта же схема действий работает для большинства других часто встречающихся справочников. Вводить указанным образом какие-либо приемы, кроме приемов справочников, не рекомендуется - генератор приемов, скорее всего, ничего полезного не предложит. Его работа в сколь-нибудь нетривиальных случаях требует наличия дополнительной разметки, истоки которой восходят к базе теорем. У теорем, создаваемых вручную, этой разметки нет.

3.2 Дизъюнкция

Как и в предыдущем случае, имеем несколько совсем простых приемов. Однако, в дополнение к ним появляется внушительных размеров пакетный нормализатор "нормили", используемый на этапе склейки фрагментов ответа задачи на описание. Большинство приемов этого пакета относятся к конкретным разделам (элементарная алгебра, дифференциальные уравнения и т.п.); их мы будем рассматривать позднее.

Приведение дизъюнкции к стандартному виду

Аналогично конъюнкции, имеются приемы для лексикографического упорядочения операндов дизъюнкции и устранения вложенных дизъюнкций.

Частичная склейка дизъюнкций

Фактически, это соотношение дистрибутивности. Теорема приема имеет вид $\forall_{abc}((a \vee b)\ \&\ (a \vee c) \leftrightarrow (a \vee (b\ \&\ c)))$. Прием усматривает две дизъюнкции в условиях задачи на описание, решаемой для редактирования найденного ответа. Если обе они не содержат неизвестных, а общая часть a их дизъюнктивных членов непуста, то выполняется группировка в одну дизъюнкцию.

Нормализатор "нормили"

Как уже говорилось выше, нормализатор необходим для завершающего редактирования ответа задачи на описание, решаемой путем разбора случаев. Только здесь и может быть предпринято исключение дизъюнктивных конструкций в ответе - например, преобразование утверждения $x = a \ \vee \ x = b$ в утверждение $x \in \{a, b\}$.

Войдем в раздел оглавления "Нормализатор НОРМИЛИ". Прежде всего, идет подменю "Справочники". Это - стандартный для всех пакетных операторов пункт, в котором собраны приемы справочников, задающие формат оператора. В нашем случае имеем два приема - справочников "быстрпреобр" и "титр". Теорема первого из них - "быстрпреобр(нормили списокпосылок неизвестные контрольтитра коррекцияпосылок уровень(5))". Она указывает, что нормализатор имеет список посылок, а число его уровней срабатывания равно 5. Нормализатору передаются в виде комментариев список неизвестных текущей задачи, а также набор (коррекцияпосылок ...). Предусмотрена выдача поясняющих текстов при обращении к нему. Второй прием определяет тот текст, который будет выдаваться на экран при обращении. Теорема его имеет вид "титр(нормили x1)". Она фиксирует переменную x1 для обозначения преобразуемого нормализатором терма (далее не используемую). При нажатии клавиши "6" появляется текст "Упростить утверждение :", который будет выдаваться на экране, если при трассировке произойдет обращение к нормализатору. Указатель приема "титр(набор(1))" констатирует, что выдается первый из фрагментов указанного выше текста (фрагменты отделяются друг от друга знаком ¬ с идущим после него номером фрагмента, и таких разделителей в нашем случае нет).

Следующий пункт оглавления - "Переключатель уровня". В нем расположен единственный прием с заголовком "окончание". Такой прием имеется во всех нормализаторах. Это не справочник, а маленький фрагмент программы нормализатора, обеспечивающий увеличение текущего уровня по исчерпании попыток применения приема на данном уровне. При достижении максимального уровня (в нашем случае - 5) прием выдает ответ - текущее преобразуемое выражение. Теорема приема такая же, как у справочника "быстрпреобр". Заметим, что перекомпиляция этого приема приведет к порче программы нормализатора - она будет попросту удалена, и придется заново компилировать весь пакет. Ввиду того, что компилятор не сможет находить удаленные ветви приемов, он при каждой повторной компиляции приема будет выходить на контрольную точку "трассировка(стоп 0)" отладчика ЛОСа. Эту точку нужно игнорировать - нажимать "0" и "Enter" для завершения компиляции. Таким образом, приемы переключателей уровня в нормализаторах лучше не трогать. Если понадобится полная перекомпиляция пакета, то нужно сначала удалить программы всех прочих приемов (через F7), затем - перекомпилировать прием переключателя уровней (лучше дважды), и лишь затем перейти к повторной компиляции основных приемов пакета.

Далее идут пункты оглавления, соответствующие собственно приемам нормализатора. Прежде всего, идут приемы, обеспечивающие простейшую логическую стандартизацию:

1. Устранение логических констант. Группа простых приемов, исключающих логические константы "истина", "ложь", которые могут возникнуть в упрощаемом утверждении либо изначально, либо после действий самого нормализатора. Например, прием $\forall_a((a \vee \text{истина}) \leftrightarrow \text{истина})$. Заметим, что так как в формате нормализатора не было символа "корень", будут отслеживаться произвольные (не только корневые) вхождения термов заданного вида в преобразуемый терм. Уровень срабатывания приемов, связанных с логическими константами, равен 1.

2. Простейшие приемы стандартизации дизъюнкций и конъюнкций: лексикографическое упорядочение операндов и устранение вложенных одинаковых операций. Аналогично приемам такого типа, применяемым при сканировании задач.

3. Совпадающие дизъюнктивные либо конъюнктивные члены. Теоремы приемов имеют вид $\forall_a((a \vee a) \leftrightarrow a)$, $\forall_a((a \,\&\, a) \leftrightarrow a)$. Уровень срабатывания 1. Программы этих приемов просматривают всевозможные пары операндов дизъюнкций и конъюнкций и сравнивают их. Оставшаяся часть операндов, не указанная в теореме приема, не изменяется.

4. Противоположные дизъюнктивные члены. Несколько простых приемов, основанных на следующих теоремах: $\forall_a((a \vee \neg a) \leftrightarrow \text{истина})$; $\forall_a b((a \,\&\, b \ \vee \ a \,\&\, \neg b) \leftrightarrow a)$; $\forall_{ab}(b - \text{число} \rightarrow ((a = b \ \vee \ \neg(a = b) \ \&\ a - \text{число}) \leftrightarrow a - \text{число}))$. Уровень

срабатывания первого приема равен 1, второго - 5, третьего - 4. В последнем случае предполагается, что a - переменная; b - некоторое выражение, для которого с помощью проверочного оператора удается усмотреть, что оно принимает числовые значения. Именно, применяется оператор "усмчисло".

5. Устранение избыточного конъюнктивного члена. Снова три простых приема; теоремы их имеют вид: $\forall_{abc}((a \vee (\neg a \vee b) \,\&\, c) \leftrightarrow a \vee c)$, $\forall_{abc}((a \,\&\, (b \vee c) \vee b) \leftrightarrow a \,\&\, c \vee b)$, $\forall_{abc}((a \,\&\, (a \,\&\, b \vee c)) \leftrightarrow a \,\&\, (b \vee c))$. Уровень срабатывания первого из них равен 1, второго и третьего - 3.

6. Вынесение за скобку общего конъюнктивного члена. Это преобразование применяется при редактировании дизъюнктивного ответа очень часто. Однако, выяснилось, что небезразличен выбор общей части двух конъюнкций, в особенности если число членов дизъюнкции больше двух. Преждевременное вынесение за скобку иногда блокировало выполнение последующих важных действий. Поэтому преобразование потребовало достаточно сложного управления, выразившегося в наличии большого числа однотипных приемов, срабатывающих на разных уровнях и выносящих за скобки разные фрагменты. Большинство приемов основано на одной и той же теореме $\forall_{abc}((a \,\&\, b \vee a \,\&\, c) \leftrightarrow (a \,\&\, (b \vee c)))$. Во-первых, прием, определяющий общую часть a как совокупность всех утверждений о типе значения объекта - "число(t)", "комплексное(t)", "Вектор(t)". Во-вторых, прием, определяющий a как совокупность всех утверждений, содержащих неизвестные, т.е. в скобках оказывется дизъюнкция известных утверждений b, c. В-третьих, прием, который определяет a как совокупность всех общих утверждений двух сравниваемых конъюнкций, но срабатывающий лишь при редактировании ответа задачи, имеющей цель "исследовать". Все эти приемы срабатывают на уровне 2. Далее идет прием, определяющий a как совокупность всех общих членов двух конъюнкций, не содержащих неизвестных. Он срабатывает на уровне 4. Наконец, идет прием, срабатывающий, если число неизвестных более одной. Тогда выбирается некоторая неизвестная x4, входящая в преобразуемое утверждение, и a определяется как совокупность всех общих членов, содержащих x4. Проверяется, что утверждение a не имеет общих неизвестных ни с b, ни с c. Уровень срабатывания этого приема тоже равен 4.

   В данном пункте имеются еще два приема. Первый из них основан на теореме $\forall_{abcd}(c\text{−число} \rightarrow ((a\text{−число} \,\&\, b \vee a = c \,\&\, d) \leftrightarrow (a\text{−число} \,\&\, (b \vee a = c \,\&\, d))))$. Уровень срабатывания его равен 3. Второй основан на теореме $\forall_{abcx}((\neg(x = a) \,\&\, b \vee \neg(x = a) \,\&\, c) \leftrightarrow (\neg(x = a) \,\&\, (b \vee c)))$. Здесь требуется, чтобы переменная x представляла собой неизвестную; утверждения b, c могут быть вырожденными. Уровень срабатывания равен 4.

   Заметим, что в некоторых теоремах приемов два утверждения вместо эквивалентности связываются знаком равенства. На самом деле, для компилятора это несущественно. В консеквенте, все же, два утверждения обычно бывают связаны не равенством, а эквивалентностью. Но в случае антецедента, выделенного указателем "идентификатор", оказалось удобным везде пользоваться не эквивалентностью, а равенством - такая запись более понятна формульному редактору.

7. Перегруппировка для конъюнктивных членов без неизвестных. Прием основан на теореме $\forall_{abcd}((a \vee b) \,\&\, c \vee a \,\&\, d \leftrightarrow a \,\&\, (c \vee d) \vee (b \,\&\, c))$. Он ориентирован

на следующую ситуацию: a, b - известны и определяют некоторые условия на параметры задачи, при которых имеется группа решений c. Для того же случая a, но уже без b, возникает еще одна группа решений - d. Тогда относящиеся к подслучаю a решения c, d объединяются вместе, а для подслучая b отдельно указываются решения c. Уровень срабатывания приема равен 3. Фильтры приема определяют указанную выше ситуацию следующим образом: b - известно; c, d - не известны и не имеют известных конъюнктивных членов.

8. Поглощение. Логическое упрощение, основанное на эквивалентности $\forall_{ab}((a \lor a \mathbin{\&} b) \leftrightarrow a)$. Уровень срабатывания равен 2.

9. Группировки для отношения принадлежности. Два приема, основанных на теоремах $\forall_{abc}(a \in (b \cup c) \leftrightarrow a \in b \ \lor \ a \in c)$, $\forall_{abc}(b = c \ \lor \ b \in a \leftrightarrow b \in a \cup \{c\})$. В обоих случаях дизъюнкция преобразуется в условие принадлежности неизвестного элемента известному множеству. Уровень срабатывания равен 2. В первом случае выражения b, c известны, a не известно; во втором случае b не известно; a, c известны.

10. Преобразование дизъюнкции равенств для неизвестной в условие принадлежности ее конечному списку. Теорема приема имеет вид $\forall_{abc}(a = b \ \lor \ b = c \leftrightarrow b \in \{a, c\})$. Выражение b не известно; a, c - известны. Имеется ряд ограничений на применение приема. Если дизъюнкция расположена внутри квантора существования, связывающего некоторые параметры выражений a, c, то преобразование блокируется, так как нарушает структуру параметрического описания. По этой же причине, оно блокируется, если имеется внешняя конъюнкция, указывающая наличие целочисленного параметра, входящего в a. Преобразование не применяется также, если текущая задача имеет цель "или", указывающую, что предпочтительнее сохранение дизъюнктивного вида ответа. Уровень срабатывания приема равен 2.

Большие серии приемов пакета относятся к редактированию ответов неравенств, тригонометрических уравнений и дифференциальных уравнений. Они будут рассмотрены в соответствующих разделах книги. Здесь же рассмотрим несколько оставшихся приемов, не вошедших в указанные серии:

1. Устранение избыточного подслучая для явно заданного параметра. Иногда дизъюнкция условий на параметры может содержать избыточные члены. Для их устранения предусмотрены два приема. Первый основан на следующей теореме: $\forall_{abcde}((a = d) \mathbin{\&} e) = c \to ((b = a \mathbin{\&} c \ \lor \ b = d \mathbin{\&} e) \leftrightarrow (b = d \mathbin{\&} e))$. Здесь b - переменная, не входящая в a, c, d, e, причем вся дизъюнкция не содержит неизвестных. Значение a переменной b, определяемое первым подслучаем, подставляется в конъюнкцию для второго подслучая. Результат $a = d \mathbin{\&} e$ упрощается, и если после упрощения возникает то же самое условие c, что и в первом подслучае, то первый подслучай можно отбросить. Чтобы выполнить упрощение и сравнение с c, служит антецедент теоремы приема. К левой его части применяется нормализатор "задача(4 тип(описать)редакция нормили цель(повторение(неизвестные))", правая часть равна c. Строго говоря, этот антецедент должен был бы иметь вид эквивалентности, так как обе его части - утверждения. Однако, как уже говорилось выше, в антецедентах теорем приемов вместо эквивалентности используется равенство.

Ограничитель "лимит(50000 x1)" нужен, чтобы не увязнуть в трудоемких случаях решения вспомогательной задачи на упрощение. Если дизъюнкция имеет много различных членов, фиксирующих значение параметра, то данный ограничитель мог бы отсечь рассмотрение части подслучаев, что нежелательно. Поэтому в него введен элемент "x1", означающий, что трудоемкость отсчитывается лишь с момента идентификации выражения a. Таким образом, прочие возможности для a останутся не отсеченными.

Второй прием основан на теореме $\forall_{abcd}(d = \text{истина} \to ((a = b \,\&\, c \;\vee\; d) \leftrightarrow d))$. Он аналогичен первому, но применяется лишь при выводе теорем, о чем свидетельствует наличии цели "редуцирование". Переменная a не входит в b, c, d. Утверждение d упрощается в предположении, что истинны утверждения $a = b$ и c. Если при этом возникает логическая константа "истина", то первый дизъюнктивный член отбрасывается. Чтобы обеспечить упрощение, к d применяется нормализатор "задача(5 упростить одз) посылки(и(равно(x1 x2)x3))". Напомним, что встречающиеся в теореме приема переменные a, b, c, d обозначены в нижележащих окнах описания приема по-другому - как x1,x2,x3,x4. Для удобства сопоставления одних обозначений другим служит переходник, прорисовываемый голубым цветом непосредственно под теоремой.

Указатель приема "копия(фикс(0 2))" защищает от обработки нормализатором то вхождение переменной d, которое размещено в консеквенте справа от эквивалентности. Разумеется, эта обработка привела бы к логически ошибочному преобразованию.

2. Усмотрение модуля. Прием основан на теореме:

$$\forall_{abcdx}(0 \le x \,\&\, 0 \le c \,\&\, 0 \le d \to ((x = \frac{c(a-b)}{d} \;\vee\; x = \frac{c(b-a)}{d}) \leftrightarrow x = \frac{c|a-b|}{d})).$$

Так как x неотрицательно, то дизъюнкция приведенных равенств сводится к одному равенству, использующему модуль выражения $a-b$. Коэффициенты c, d введены в прием для усмотрения преобразования в возможно более общей ситуации. Нетрудно продолжить обобщение, перейдя от выражений $a-b$, $b-a$ к их степеням, имеющим рациональный показатель с нечетными знаменателем и числителем. Это упражнение оставляется читателю. Заметим, что многие приемы решателя без труда могут быть обобщены - настоящая его версия, скорее, является коллекцией наглядных пособий для анализа логических процессов, чем программным продуктом, ориентированным на пользователя.

Отметим наличие в приеме указателя "нормзнака(x2 минус плюс)". Выражение a будет идентифицировано из рассмотрения какого-то слагаемого суммы $b-a$, имеющего знак минус; остальные слагаемые данной суммы составят b. Указатель "нормзнака(...)" позволяет, рассматривая сумму $a-b$, идентифицировать с $-b$ результат изменения всех знаков слагаемых в уже найденной сумме для b.

3. Исключение модуля либо сигнума. В подразделе собраны три приема, позволяющие исключать модуль либо сигнум из дизъюнкции. Первый прием основан на теореме

$$\forall_{abcx}(x = \frac{a|b|}{c} \;\vee\; x = -\frac{a|b|}{c} \leftrightarrow x = \frac{ab}{c} \;\vee\; x = -\frac{ab}{c}).$$

Уровень срабатывания его равен 1. Второй прием основан на теореме

$$\forall_{abcdx}(x = -\frac{a + c\mathrm{sg}(b)}{d} \ \vee \ x = \frac{c\mathrm{sg}(b) - a}{d} \leftrightarrow x = -\frac{a+c}{d} \ \vee \ x = \frac{c-a}{d}).$$

Уровень срабатывания здесь равен 3. Третий прием представляет собой обобщение первого на более сложные случаи размещения модулей в дизъюнкции. Его теорема имеет вид

$$\forall_{abcdef}(a = c \ \& \ a = d \ \& \ b = e \ \& \ b = f \to a \ \vee \ b \ \vee \ i \leftrightarrow c \ \vee \ d \ \vee \ i).$$

Только по виду теоремы, без учета прочих компонент описания приема, понять ничего нельзя. Смотрим в четвертое окно описания и обнаруживаем указатель "контекст(позиция(х7 х1)вид(х7 модуль(х8)))". Он означает, что внутри дизъюнктивного члена a (т.е. х1) расположен модуль некоторого выражения х8. В третьем окне находим фильтр "вхождениетерма(х2 модуль(х8))", означающий, что тот же самый модуль встречается и внутри дизъюнктивного члена b. По виду антецедентов ясно, что они используются для обработки выражений a, b нормализаторами. Обращаемся к первому антецеденту и находим, что выражение c получено из a применением нормализатора "норм(замечание(модуль х8)) посылки(меньше(0 х8))". Очевидно, это - попытка раскрыть модуль в предположении, что под ним стоит положительное выражение. Аналогичным образом просматриваем прочие антецеденты. Оказывается, что c, f - результаты раскрывания модуля в выражениях a, b при положительном либо отрицательном х8; d, e - результаты раскрывания модуля в тех же выражениях при противоположных знаках х8 (соответственно, отрицательном либо положительном). Фильтры "равно(терм(х3)терм(х6))" и "равно(терм(х4)терм(х5))" означают, что выражения a, b после раскрывания модулей с противоположными знаками для х8 и упрощения становятся идентичными. Это и дает основание заменить их в дизъюнкции на выражения c, d.

4. Усмотрение частного случая, подпадающего под общее решение. Здесь представлено несколько приемов, позволяющих отбрасывать один из членов дизъюнкции, определяющий такие значения неизвестной, которые уже учтены в остальных членах. Начнем с приема, теорема которого имеет вид:

$$\forall_{abcf}(f(b) = c \to ((\neg(a = b) \ \& \ x = f(a) \ \vee \ a = b \ \& \ x = c) \leftrightarrow x = f(a)).$$

В ответе различаются два случая: при значении b параметра a неизвестная x равна c; при прочих значениях параметра a она равна $f(a)$. Антецедент реализует обращение к нормализатору "задача(4 упростить)", обрабатывающему выражение $f(b)$. Если после упрощения получается выражение c, то выделение частного случая $a = b$ не нужно - общий случай дает такое же выражение для x. Выражение $f(a)$, согласно указателю "отображение(х6)" (f - это х6), идентифицируется с произвольным термом. Чтобы получить $f(b)$, в данный терм вместо переменной, идентифицированной с a, подставляется терм, идентифицированный с b. Уровень срабатывания приема равен 5.

Следующий прием основан на теореме

$$\forall_{xabf}(f(a) \to ((x \in \{a;b\} \ \vee \ f(x)) \leftrightarrow x \in \{;b\} \ \vee \ f(x))).$$

Здесь неизвестная x либо удовлетворяет условию $f(x)$, либо принадлежит конечному списку, в котором выделяется элемент a, причем через b обозначен набор остальных элементов списка. Если просмотреть теорему приема в текстовой записи (клавиша "т"), то обнаружится, что обозначения $\{a;b\}$ и $\{;b\}$ расшифровываются как "перечень(префикс(x1 x2))"и "перечень(x2)" соответственно. Чтобы a идентифицировалось не с первым, а с произвольным элементом набора, введен указатель "список(фикс(0 1 1 2 1))". Указатель вхождения "фикс(...)" выделяет вхождение в теорему подвыражения "префикс(x1 x2)". Антецедент теоремы имеет вид $f(a)$, т.е. получен из условия "общего случая" $f(x)$ подстановкой в него a. Указатель "легковидеть(1)" определяет проверку данного антецедента с помощью вспомогательной задачи на доказательство, имеющей уровень обращения 4. Чтобы предотвратить неоправданно большие затраты времени на проверку, введен ограничитель "лимит(30000)". Кроме того, отброшены случаи, когда условие $f(x)$ имеет вид равенства либо принадлежности. Как мы помним, для таких случаев имеются свои приемы упрощения дизъюнктивного ответа. В заменяющей части теоремы случай $x = a$, оказавшийся включенным в серию значений $f(x)$, отброшен из конечного списка.

Теорема следующего приема совсем простая:

$$\forall_{xaf}(f(a) \to ((x = a \ \vee \ f(x)) \leftrightarrow f(x))).$$

В антецеденте устанавливается, что особый случай $x = a$ для неизвестной x подпадает под некоторое условие $f(x)$, определяемое другим дизъюнктивным членом.

Несколько представленных в данному пункте приемов предназначены для редактирования решений дифференциальных уравнений - в них вместо обычной неизвестной встречается функциональная неизвестная $y(x)$. Эти приемы сейчас разбирать не будем.

Оставшиеся приемы аналогичны уже разобранным. Ограничимся тем, что приведем несколько их теорем:

$$\forall_{xaf}(f(a) \to ((x = a \ \vee \ \neg(x = a) \ \& \ f(x)) \leftrightarrow f(x))),$$

$\forall_{xabfg}(a - \text{число} \ \& \ f(a) \to ((x \in \{a;b\} \ \vee \ x - \text{число} \ \& \ (f(x) \ \vee \ g(x))) \leftrightarrow x \in \{;b\} \ \vee \ x - \text{число} \ \& \ (f(x) \ \vee \ g(x)))),$

$\forall_{abcfg}(f(b) = c \to ((\neg(a = b) \ \& \ x = f(a) \ \vee \ a = b \ \& \ x = c \ \& \ g) \leftrightarrow (\neg(a = b) \ \vee \ g) \ \& \ x = f(a))),$

$\forall_{abcf}(f(b) = c \to ((\neg(a - b = 0) \ \& \ x = f(a) \ \vee \ a = b \ \& \ x = c) \leftrightarrow x = f(a))).$

3.3 Равенство

В разделе содержится несколько простых приемов, используемых в редко встречающихся случаях. Кроме того, имеется нормализатор "нормравно", используемый также достаточно редко.

Устранение равенств, вводящих не используемые обозначения

Теорема приема имеет вид $\forall_{ab}(a = b \leftrightarrow$ истина). Здесь используется чисто технический трюк - для исключения утверждения из списка посылок оно заменяется на константу "истина". Фильтры приема определяют следующий контекст его срабатывания: рассматриваемое равенство является посылкой задачи на исследование, имеющей цель "противоречие" (т.е. решаемой для установления противоречивости списка посылок). a - переменная, не входящая в b и не встречающаяся в прочих посылках задачи.

Преобразование отрицания числового равенства с помощью вспомогательной задачи

Если отрицание числового равенства является условием задачи на описание, то чаще всего оно используется для сопровождения по о.д.з. В этой ситуации, даже если равенство содержит неизвестные, нецелесообразно его относительно них разрешать. Однако, в некоторый момент, наступающий ближе к окончанию решения, отрицание равенства может оказаться не используемым для сопровождения. Тогда применяется прием, обращающийся к вспомогательной задаче на описание для явного разрешения такого равенства. Оно разрешается относительно всех встречающихся в нем неизвестных. Теорема приема имеет вид: $\forall_{abc}(c = (a = b) \rightarrow (\neg(a = b) \leftrightarrow \neg c))$. Антецедент организует обращение к вспомогательной задаче на описание, условием которой служит $a = b$. Переменной c присваивается ответ задачи, и отрицание равенства заменяется на отрицание c. Фильтры приема уточняют следующий контекст срабатывания: отрицание равенства является условием задачи на описание и не используется для сопровождения по о.д.з. Выражение a принимает числовые значения; равенство содержит неизвестные и не имеет вида $x = t$, где x - неизвестная, t - известно. Число неизвестных задачи не более одной. Если равенство содержит тригонометрическое подвыражение с неизвестными, то в задаче не осталось ни одного уравнения. К этому добавлены ограничения, относящиеся к функциональным (фактически - дифференциальным) уравнениям. Указатель "примечание(условие(или(входит(существует x3)входит(длялюбого x3))серия))" проверяется наличие квантора в утверждении c. Если такой квантор есть, то $\neg c$ снабжается комментарием "серия", означающим, что условие может рассматриваться как описание серии значений неизвестной. Так как после явного разрешения равенства некоторые из сопровождающих его по о.д.з. неравенств могут оказаться не нужными, веса всех неравенств понижаются до 1 - для повторного их рассмотрения.

Нормализация дизъюнктивного условия с отрицанием равенства

Прием основан на теореме $\forall_{abc}(\neg(a = b) \ \vee \ c \leftrightarrow \neg(a = b) \ \vee \ a = b \ \& \ c)$. Здесь a - переменная, не входящая в b, но входящая в c. Утверждение c идентифицируется как дизъюнкция всех оставшихся членов. Присоединение к ней равенства $a = b$ позволяет зафиксировать значение переменной a - параметра либо неизвестной. В обоих случаях c приобретает, после возможных упрощений, более явный вид. Прием срабатывает на уровне 0. Требуется, чтобы дизъюнкция представляла собой условие задачи на описание, содержащее неизвестные, а утверждение c не имело конъюнктивного члена, определяющего значение переменной a.

Устранение несущественной неизвестной, явно определяемой равенством

В подразделе представлены три приема, отличающихся лишь типом значений устраняемой неизвестной. Рассмотрим один из них - например, имеющий следующую теорему:

$$\forall_{abc}(b - \text{целое} \to \exists_x(x - \text{целое} \,\&\, (a \,\&\, x = b \ \vee \ c)) \leftrightarrow a \ \vee \ c).$$

Заголовок приема - "связка". Это означает, что происходит исключение несущественной неизвестной x, т.е. такой неизвестной, которая упомянута в цели (параметры ...) текущей задачи на описание. Конъюнктивные члены подкванторного утверждения суть все условия задачи, содержащие x. В нашем случае таких условия должно быть два - "целое(x)" и дизъюнкция $a \,\&\, x = b \ \vee \ c$. Фильтры приема доопределяют следующий контекст срабатывания: неизвестная x не входит ни в a, ни в b, ни в c. Антецедент теоремы определяет проверку целочисленности выражения b. Если эта проверка удалась, то существование x очевидно, так что можно отбросить два указанных выше условия, заменив их на дизъюнкцию $a \ \vee \ c$. Хотя неизвестная x после применения приема в условиях задачи уже не встречается, в списке неизвестных она будет сохранена.

Уровень срабатывания приема равен 0. Другие два приема относятся к типам значений "натуральное" и "число".

Ориентация равенства

В подразделе представлено несколько приемов, дополняющих описанный выше общий прием ориентации равенства. Теоремы их - либо вида $\forall_{ab}(b - \text{число} \to a = b \leftrightarrow b = a)$, либо вида $\forall_{ab}(a = b \leftrightarrow b = a)$. Остальное доопределяется фильтрами и указателями приема.

Например, задается следующий контекст срабатывания: рассматривается корневое равенство задачи на исследование, имеющей цель "известно". Это - обычная ситуация для планиметрических задач на вычисление, задач по аналитической геометрии, физике, и т.д. b - неизвестная задачи, не входящая в a. Все параметры выражения a - числовые, причем это выражение не однобуквенное. Указатель "коммутативно(фикс(0 1))" уточняет, что перестановки частей равенства при идентификации не происходит - т.е. неизвестная b расположена в правой части. После перенесения неизвестной в левую часть прием сопровождает равенство комментарием "ориентацияравенства", блокирующим встречную перестановку.

Определим контекст срабатывания еще одного приема такого типа. Здесь равенство не содержит переменных, причем выражение a - почти стандартная константа, т.е. может содержать лишь операции умножение и "величина" (последняя собирает десятичную запись из цифр). Выражение b - более сложное, т.е. не имеет указанного вида. Тогда оно переносится в левую часть.

Нормализатор "нормравно"

Пропустив несколько простых приемов, используемых в очень редких случаях, переходим к нормализатору "нормравно". Он служит для общей стандартизации равенств и применяется редко - всего в одном-двух десятках приемов. Нормализатор содержит, во-первых, прием, заменяющий на константу "истина" равенства с совпадающими частями. Теорема его имеет вид $\forall_a(a = a)$. Если не сопроводить такую теорему специальным указателем, прием просто будет заменять произвольное выражение a само на себя, и нормализатор зациклится. Однако, указатель "эквивалентно" поясняет компилятору, что заменяться должно само равенство. Так как заменяющий терм не указан, то в начале компиляции теорема будет преобразована к виду $\forall_a((a = a) \leftrightarrow \text{истина})$.

Во-вторых, нормализатор имеет прием, заменяющий равенство либо его отрицание на константу "истина", если в контексте замены уже имеется такое равенство либо отрицание.

Далее идут три специальных приема. Первые два заменяют равенства наборов на конъюнкции равенств их элементов; рассматриваются одноэлементные наборы и пары. Третий - относится к элементарной алгебре. Если равенство имеет вид $ab = ac$, то он проверяет отличие множителя a от нуля и сокращает на него. Он выпадает из разряда "общелогических" и был занесен в нормализатор, чтобы в нескольких примерах сэкономить на обращениях к более мощным нормализаторам равенств (например, таким, как "нормчисло"). Последние обращались к дорогостоящей процедуре разложения на множители, в то время как достаточно было просто сократить явно обозначенные общие множители. Видимо, этот и другие аналогичные "случайные" приемы при развитии решателя подлежат пересмотру. В данном случае можно было бы обращаться к нормализатору "нормчисло" с блокировкой трудоемких действий, либо ввести еще один упрощенный нормализатор числовых равенств.

3.4 Условное выражение

Здесь собрано сравнительно большое число приемов, часто используемых при работе с условными выражениями. Напомним, что такие выражения имеют вид "вариант(A t_1 t_2)", где A - условие, t_1 - выражение, выбираемое при истинном условии, t_2 - выражение, выбираемое при ложном условии.

Условие - логическая константа

Если условие под "вариант"ом представляет собой логическую константу, то условное выражение исключается. Теоремы приемов имеют вид $\forall_{ab}((a \text{ при ложь, иначе } b) = b)$, $\forall_{ab}((a \text{ при истина, иначе } b) = a)$.

Равенство альтернативных выражений

Если альтернативные выражения условного выражения равны, то это условное выражение исключается. Если имеются два вложенных друг в друга условных выражения, у которых какие-то два из альтернативных выражений совпали, то они преобразуются в одно условное выражение. В первом случае теорема приема имеет вид $\forall_{ab}((b \text{ при } a, \text{иначе } b) = b)$. Во втором случае имеется несколько теорем, для различных случаев размещения совпадающих подвыражений. Они аналогичны, и мы ограничимся примером одной такой теоремы:

$$\forall_{abcd}((b \text{ при } a, \text{иначе } (d \text{ при } c, \text{иначе } b)) = (b \text{ при } a \ \vee\ \neg c, \text{иначе } d)).$$

В ряде специальных случаев переход от двух вложенных условных выражений к одному нежелателен, так как нарушает стандартный вид описаний, принятый в приемах предметной области. Например, это относится к задачам на вычисление определителей "общего вида". Здесь указанный переход блокируется.

Упрощение дизъюнктивного условия

Чтобы упрощать условие условного выражения, при обучении возникло несколько простых приемов. Один из них - усмотрение избыточности дизъюнктивного члена, наподобие реализованному в пакете "нормили". Теорема приема имеет вид:

$$\forall_{xabcf}(f(a) \to (b \text{ при } x = a \ \vee \ f(x), \text{иначе } c) = (b \text{ при } f(x), \text{иначе } c)).$$

Проверка антецедента $f(a)$ предпринимается, если x - переменная, встречающаяся в утверждении $f(x)$; a - константа.

Отрицание равенства варианта одному из его подвыражений

Отрицание равенства условного выражения одному из его альтернативных подвыражений легко приводится к виду с исключенным условным выражением. Для этого используются два приема, имеющие следующие теоремы:

$$\forall_{abc}(\neg((a \text{ при } b, \text{иначе } c) = a) \leftrightarrow \neg b \ \& \ \neg(a = c)),$$

$$\forall_{abc}(\neg((a \text{ при } b, \text{иначе } c) = c) \leftrightarrow b \ \& \ \neg(a = c)).$$

Уровень срабатывания приемов равен 0.

Условное выражение, выделяющее избыточный подслучай

Если условное выражение имеет вид $(a \text{ при } x = y, \text{иначе } f(x))$, причем $f(y)$ допустимо по о.д.з. и равно a, то это выражение заменяется на $f(x)$. Теорема приема имеет вид

$$\forall_{xyaf}(f(y) = a \ \& \ \text{одз} f(y) \to ((a \text{ при } x = y, \text{иначе } f(x)) = f(x))).$$

Уровень срабатывания равен 2. Конструкция "одз($f(y)$)" в антецеденте воспринимается компилятором как указание на обращение к процедуре "Одз", определяющей о.д.з. выражения $f(y)$. Указатель приема "легковидеть(2)" определяет проверку найденных условий на о.д.з. с помощью вспомогательной задачи на доказательство. Первый антецедент определяет упрощение выражения $f(y)$ с помощью вспомогательной задачи на преобразование и сравнение результата с a. Чтобы упрощение не выполнялось до того, как будет установлена корректность выражения $f(y)$, введены указатели "конец(1)" и "начало(2)". В этом же направлении работает и указатель "копия(фикс(2 1))" - он говорит, что при обращении к процедуре "Одз" должен использоваться не результат упрощения выражения $f(y)$, а само это выражение. Для срабатывания приема необходимо, чтобы x представляло собой переменную. Однако, указатель "заменатермов(фикс(0 1 3))", в принципе, позволяет отказаться от такого требования. Он говорит, что выражение $f(y)$ получается из идентифицированного с $f(x)$ выражения T заменой всех вхождений терма x (не обязательно переменной) на терм y. Случай переменной x можно было бы обработать и проще - используя указатель "отображение(х6)".

Условие, фиксирующее значение переменной

Если условие имеет вид $x = a$, то предпринимается подстановка a вместо x в первое альтернативное выражение. Теорема приема здесь такова:

$$\forall_{xab}((f(x) \text{ при } x = a, \text{иначе } b) = (f(a) \text{ при } x = a, \text{иначе } b)).$$

Проверяется, что x - переменная, входящая в выражение $f(x)$. Заметим, что сама по себе запись $f(x)$ вовсе не означает, что данное выражение обязательно содержит x. Она лишь означает, что новые термы вида $f(t)$ будут строиться из идентифицированного с $f(x)$ терма T путем подстановки вместо переменной, идентифицированной с x, выражения, построенного по шаблону t. Далее, прием проверяет, что a - не содержащее x и отличное от переменной выражение. Уровень срабатывания его равен 0.

Подстановка значения согласно равенству - конъюнктивному члену условия

Если условное выражение имеет вид (a при $t = b$ & d, иначе c), причем b - константа, t - неконстантное выражение, входящее в a, то выполняется замена всех вхождений t в a на b. Теорема приема имеет вид

$$\forall_{abcdft}(a = f(t) \rightarrow ((a \text{ при } t = b \;\&\; d, \text{иначе } c) = (f(b) \text{ при } t = b \;\&\; d, \text{иначе } c))).$$

Ее антецедент нужен для определения по выражению a функционального шаблона $f(t)$. Согласно указателю "заменатермов(фикс(1 2))", этот шаблон представляет собой непустой список вхождений выражения t в a. Терм $f(b)$ в заменяющей части теоремы получается путем замены всех вхождений шаблона на выражение b. Нормализаторы приема задают обращение к вспомогательной задаче на упрощение данного терма.

Исключение отрицания в условии

Если условие имеет вид отрицания, то это отрицание отбрасывается, а альтернативные выражения переставляются.

Разбор случаев в задаче на описание

Если условное выражение встречается в условиях задачи на описание, то предпринимается разбор случаев по условию a данного выражения. Для этого создано несколько приемов; каждый из них добавляет к списку условий дизъюнкцию $a \vee \neg a$. Дизъюнкция сопровождается комментарием "разборслучаев", ускоряющим переход к рассмотрению альтернатив. Теоремы всех приемов имеют вид $\forall_a(a \vee \neg a)$. Приемы снабжены указателями "контрольвывода(вариант(x1 x2 x3))", означающими, что инициализация попытки применения происходит при обнаружении где-либо в задаче условного выражения "вариант(a b c)". Контекст срабатывания предполагает, что условное выражение обнаружено в условии задачи на описание, не имеющей условий с заголовком "или". Перечислим оставшиеся компоненты контекстов срабатывания рассматриваемых приемов:

1. Условное выражение содержит неизвестные. Уровень срабатывания равен 2.
2. Условное выражение A не содержит неизвестных. Либо текущее условие задачи имеет комментарий "упрощвариант", либо A расположено внутри утверждения $b \in A$, причем хотя бы одно из альтернативных выражений b, c имеет своим заголовком символ "прямоепроизведение". Уровень срабатывания равен 2.
3. Условное выражение не содержит неизвестных. Оно размещено внутри условия принадлежности неизвестной некоторому выражению. Уровень срабатывания равен 6.

Разбор случаев в задаче на доказательство либо преобразование

Ситуация аналогична случаю задач на описание: обнаруживается условное выражение "b при a, иначе c", и выводится посылка $a \ \vee \neg a$, снабжаемая комментарием "разборслучаев". Отдельно рассмотрена ситуация, когда a есть строгое неравенство $d < e$. Тогда выводится посылка $d < e \ \vee \ e \leq d$. Имеется ряд ограничений на срабатывание: отбрасываются задачи, целевая установка которых определяет ускоренный режим; проверяется отсутствие переменных выражения a, связанных внешними кванторами и описателями, и т.п.

Усмотрение истинности либо ложности неравенства

Если в условии условного выражения встречается неравенство, то предпринимаются попытки установить его истинность либо ложность с помощью проверочных операторов. В качестве примера приведем две теоремы приемов: $\forall_a(0 \leq a \to (b$ при $a < 0$, иначе $c) = c)$ и $\forall_{abcd}(0 \leq a \to (c$ при$(0 \leq a \ \vee \ b)$, иначе $d) = c)$. Конечно, чтобы исключать условные выражения, можно было бы создать приемы, обращающиеся для проверки либо опровержения их условий к вспомогательным задачам на доказательство. Однако, такой вариант существенно замедлил работу решателя, и в итоге пришлось ограничиться сравнительно быстрыми обращениями к проверочным операторам. Для организации их создаются серии приемов, соответствующие различным отношениям, обеспеченным указанными операторами. Эти приемы выполняют, по существу, одно и то же - проверку истинности либо ложности условия. Если простыми средствами устранить условное выражение не удается, оно обычно порождает разбор случаев. Если условие выражения было истинным либо ложным, то при разборе случаев данный факт в конце концов выясняется. Таким образом и происходит исключение условных выражений в "неочевидных" ситуациях.

Преобразование неравенства в равенство

Если условием условного выражения служит строгое неравенство $a < b$, причем из контекста ясно, что истинно $a \leq b$, то условие заменяется на $a = b$, а альтернативные выражения меняются местами. Для экономии пришлось ограничиться лишь непосредственным усмотрением неравенства $a \leq b$ в контексте условного выражения. Теорема приема имеет вид $\forall_{abc}(0 \leq a \to (b$ при $0 < a$, иначе $c) = (c$ при $a = 0$, иначе $b))$. Аналогичный прием создан для случая, когда ноль находится в правой части неравенства. Кроме того, имеется прием для усмотрения нестрогого неравенства из условия принадлежности: $\forall_{abcm}(b \in \{m, \ldots, n\} \to (a$ при $m < b$, иначе $c) = (c$ при $b = m$, иначе $a))$.

Усмотрение ложности равенства

Если условие условного выражения представляет собой числовое равенство, то для усмотрения его отрицания используется проверочный оператор "усмне0". Он рассчитан на проверку утверждений вида $\neg(a = 0)$. Первые два приема, у которых условное выражение имеет вид (b при $a = 0$, иначе c) и (a при $(b = 0 \vee c)$, иначе d), рассчитаны на числовые равенства с нулем в одной части и срабатывают на уровне 0. Третий прием, предназначенный для числовых равенств общего вида, срабатывает на уровне 2. Во всех этих случаях введены достаточно сильные ограничения на трудоемкость проверки.

Равенство условного выражения нулю

Если удается усмотреть отличие одного из альтернативных выражений от 0, то легко упростить утверждение, имеющее вид равенства условного выражения нулю. Например, для первого из альтернативных выражений теорема приема такова: $\forall_{abc}(\neg(a = 0) \to ((a \text{ при } b, \text{иначе } c) = 0) \leftrightarrow (\neg b \ \& \ c = 0))$. Антецедент обрабатывается проверочным оператором.

Условие принадлежности двухэлементному множеству

Если даны вложенные условные выражения, причем условие внешнего имеет вид принадлежности некоторого a двухэлементному множеству $\{b, c\}$, а условие внутреннего - вид равенства a и b, то происходит переход к паре условий $a = b$ и $a = c$.

Отбрасывание избыточного подслучая

Рассматривается случай трех вложенных условных выражений, в котором одно из альтернативных выражений оказывается недостижимым. Теорема приема имеет вид $\forall_{ABabcd}((a \text{ при } (\neg A \ \& \ \neg B), \text{иначе } (b \text{ при } A, \text{иначе } (c \text{ при } B, \text{иначе } d))) = (a \text{ при } (\neg A \ \& \ \neg B), \text{иначе } (b \text{ при } A, \text{иначе } c)))$. Указатель "отрицание(не x26 x27)" позволяет произвольным образом относить при идентификации отрицание к утверждениям A, B либо $\neg A, \neg B$.

Сумма условных выражений

При задании матриц переменного порядка часто используются условные выражения. Если вычисляется определитель такой матрицы, то при вычитании строк или столбцов могут появиться суммы условных выражений, которые необходимо преобразовать к виду условного выражения. Для этого предусмотрен прием, основанный на теореме:

$\forall_{abcdPQ}((a \text{ при } P, \text{иначе } b) + (c \text{ при } Q, \text{иначе } d) = ((a + c \text{ при } Q, \text{иначе } a + d) \text{ при } P, \text{иначе } (b + c \text{ при } Q, \text{иначе } b + d)))$.

Усмотрение избыточности отрицания равенства

Если в условии условного выражения оговаривается, что переменная x не принимает некоторого значения a, причем при $x = a$ оба альтернативных выражения совпадают, то ограничение $\neg(x = a)$ отбрасывается.

Выделение частного случая кванторной импликации

Если в консеквенте кванторной импликации по переменной x встречается условное выражение с условием $x = t$, а значение t удовлетворяет антецедентам, то импликация преобразуется в конъюнкцию двух утверждения. Первое относится к частному случаю $x = t$, второе - получается из исходной импликации добавлением антецедента $\neg(x = t)$ и заменой условного выражения на его второе альтернативное выражение.

3.5 Нормализатор "нормлог"

Чтобы упрощать логическую структуру утверждений, создаваемых приемом ГЕНОЛОГа, обычно используется нормализатор "нормлог". К нему полезно обращаться, если создаваемое приемом утверждение имеет логические связки или кванторы, а его фрагменты, после применения к ним нормализаторов, могут оказаться логическими константами.

Нормализатор "нормлог" содержит лишь самые простые приемы, относящиеся к логическим константам, логическим связкам, равенству и кванторам. Прежде всего, исключаются случаи появления логической константы под логической связкой либо квантором. Отрицание конъюнкции либо дизъюнкции преобразуется в дизъюнкцию либо конъюнкцию отрицаний; двойное отрицание устраняется. Выполняются упрощения при появлении совпадающих либо противоположных операндов конъюнкции или дизъюнкции. Устраняются вложенные конъюнкции либо дизъюнкции.Используются эквивалентности $\forall_{ab}(a \,\&\, (b \vee a) \leftrightarrow a)$ и $\forall_{ab}(a \,\&\, (b \vee \neg a) \leftrightarrow a \,\&\, b)$. Исключаются равенства с совпадающими частями. Выносится за скобку общий дизъюнктивный член двух конъюнкций. Наконец, усматривается ложность конструкций вида $\forall_x(\neg f(x) \,\&\, \exists_x(f(x) \,\&\, g(x)))$.

3.6 Нормализатор "нормодз"

Нормализатор "нормодз" используется процедурой "одз", пополняющей задачу условиями на область допустимых значений. Он лишь формально рассматривается как общелогический нормализатор. Реально аппарат сопровождения по о.д.з. востребован лишь в элементарной алгебре. Соответственно, к этому разделу относятся и все приемы нормализатора. Нормализатор существенно ускоряет решение многих задач, разбивая изначально громоздкие условия неотрицательности, отрицания навенства нулю и т.п., на множество мелких условий тех же типов. Эти действия вынесены за рамки сканирования задачи. Если их выполнять в процессе сканирования, то начальный этап решения будет перегружен мелкими упрощающими преобразованиями сопровождающих утверждений. Приведем здесь лишь несколько типичных примеров приемов нормализатора. Заметим, что нормализатор "нормодз" - корневой, т.е. заменяемая часть приема идентифицируется со всем преобразуемым утверждением.

Отличие произведения от нуля

Прием преобразует отрицание равенства произведения нескольких сомножителей нулю в конъюнкцию отрицаний равенств нулю этих сомножителей. Теорема его имеет вид $\forall_{ab}(\neg(ab = 0) \leftrightarrow \neg(a = 0) \,\&\, \neg(b = 0))$. Чтобы прием срабатывал не только в случае, когда число сомножителей равно 2, он снабжен указателем "набор(первыйтерм)". Указатель "вариант(фикс(0 1 1 1)Умножение)" обобщает прием на случай комплекснозначного произведения. Отрицания равенств нулю сомножителей, вообще говоря, могут требовать дальнейшего упрощения. Например, это происходит, если имеются сомножители степенного вида. Поэтому данные отрицания сопровождены нормализаторами "нормодз".

Отличие дроби от нуля

Условие отличия дроби от нуля преобразуется в условие отличия от нуля числителя этой дроби. Так как обрабатывается система утверждений, обеспечивающих о.д.з., то условие отличия знаменателя дроби от нуля учтено в других утверждениях.

Устранение имеющего известный знак множителя ненулевой части неравенства

Если рассматривается неравенство с нулевой частью, причем ненулевая часть имеет вид произведения, то предпринимаются попытки усмотреть строго положительные либо строго отрицательные множители и отбросить их (с соответствующей перестановкой частей неравенства). Например, для отбрасывания положительных множителей правой части служит прием: $\forall_{ab}(0 < a \to 0 \leq ab \leftrightarrow 0 \leq b)$. Проверка антецедента выполняется проверочным оператором. Ограничитель "лимит(20000)" делает эту проверку быстрой.

3.7 Квантор существования

Основным объектом данного раздела является нормализатор "нормсуществует". Как и нормализатор "нормили", он применяется для завершающего редактирования ответов задач на описание. Необходимость в обращении к нормализатору возникает для ответов, имеющих вид параметрических описаний (см. реализованные на ЛОСе приемы квантора "существует"). Лишь небольшая часть приемов нормализатора имеет общелогический характер; прочие относятся к конкретным разделам. В первую очередь, это большая группа приемов, используемых для редактирования ответов дифференциальных уравнений. Во-вторых, существенно меньшая, но тоже значительная группа приемов, используемых при редактировании ответов тригонометрических уравнений и неравенств. В этом разделе мы ограничимся только общелогическими приемами нормализатора "нормсуществует". Заметим, что нормализатор не является корневым - заменяемый терм приема может иметь произвольные вхождения внутри преобразуемого нормализатором утверждения.

Спуск квантора существования на дизъюнктивные операнды

Если под квантором существования расположена дизъюнкция нескольких утверждений, в каждом из которых выделяется группа не зависящих от переменных кванторной приставки конъюнктивных членов, то квантор преобразуется в дизъюнкцию. Теорема приема (для случая двух дизъюнктивных членов) имеет вид:

$\forall_{fghAB}(\exists_n((A \ \& \ f(n) \ \vee \ B \ \& \ g(n)) \ \& \ h(n)) \leftrightarrow A \ \&\exists_n(f(n) \ \&h(n)) \ \vee \ B \ \& \ \exists_n(g(n) \ \& \ h(n)))$.

Указатель "кортежпеременных(x14)" означает, что переменная n будет идентифицироваться со связывающей приставкой произвольной длины; указатель "внешнийквантор(фикс(0 1))" - что при идентификации не предпринимается попытка представить квантор общности как отрицание квантора существования. Указатель "единица(истина x8)" означает, что внешняя по отношению к дизъюнкции часть $h(n)$ может отсутствовать. Утверждения A, B невырождены (нет указателя, допускающего возможность их вырождения) и не содержат переменных кванторной приставки n. Идентифицируются они именно по данному признаку - как конъюнкция всех членов, не зависящих от n.

Аналогичный прием имеется для случая дизъюнкции длины 3.

Исключение переменной, явно определенной при помощи равенства

Рассматривается два случая: $\forall_{abc}(\exists_x(x = a \ \& \ b(x) \ \vee \ c(x)) \leftrightarrow b(a) \ \vee \ \exists_x(c(x)))$ и $\forall_{fg}(\exists_{xy}(x = f(y) \ \& \ g(x,y)) \leftrightarrow \exists_y(g(f(y),y)))$. Во втором случае y идентифицируется со списком переменных произвольной длины (в том числе, с пустым списком).

Кроме того, имеется прием для исключения квантора существования, основанный на теореме $\forall_a(\exists_x(x = a))$. Он заменяет квантор существования на константу "истина".

Вынесение из-под квантора условий, не зависящих от переменных его связывающей приставки

Рассматриваются два случая: $\forall_{ab}(\exists_x(a(x) \ \vee \ b) \leftrightarrow \exists_x(a(x)) \ \vee \ b)$ и $\forall_{abc}(\exists_x(a(x) \ \& \ b \ \vee \ c(x)) \leftrightarrow b \ \& \ \exists_x(a(x)) \ \vee \ \exists_x(c(x)))$. Здесь x идентифицируется со списком переменных произвольной длины.

Устранение вложенности кванторов существования

Если конъюнктивным членом утверждения под квантором существования является другой квантор существования, то предпринимается слияние его с внешним квантором. Теорема приема имеет вид: $\forall_{ab}(\exists_x(\exists_y(a(x,y)) \ \& \ b(x)) \leftrightarrow \exists_{xy}(a(x,y) \ \& \ b(x)))$. x, y идентифицируются со списками переменных произвольной длины.

Дополнительные преобразования

Как уже говорилось выше, нормализатор "нормсуществует" не корневой. Его приемы могут нарушать структуру стандартизированных логических конструкций. Для восстановления такой структуры в нормализатор введены некоторые сопутствующие приемы. Например, имеются приемы для устранения вложенных дизъюнкций или конъюнкций, а также для лексикографического переупорядочения их операндов. Кроме того, есть приемы для исключения логических констант.

3.8 Квантор общности

В разделе представлены несколько простых приемов, а также нормализатор "нормдлялюбого". В отличие от нормализатора "нормсуществует", он применяется редко и содержит лишь общелогические приемы.

Попытка усмотреть истинность кванторного условия задачи на описание

Квантор существования может встретиться в ответе задачи на описание для задания серии значений неизвестных. Квантор общности тоже может встретиться в ответе - для указания серии запрещенных значений. Обычно он имеет вид $\forall_n(n$ – целое $\rightarrow \neg(a = f(n)))$. Чтобы усматривать избыточность такого рода ограничения, введен прием, который обращается к вспомогательной задаче на доказательство утверждения $\neg(a = f(n))$ в предположении "n – целое". Прием применяется лишь при наличии у задачи цели "учетответа", указывающей на редактирование параметрического описания.

Контрапозиция для перенесения неизвестной в консеквент

Ранее встречались приемы, применимые только к таким кванторным импликациям в условиях задач на описание, которые не имеют неизвестных в антецеденте. Поэтому бывает полезно перенести все неизвестные утверждения в консеквент. Если утверждение с неизвестными единственное, можно воспользоваться приемом, основанным на контрапозиции: $\forall_{fgh}(\forall_x(f(x)\ \&\ g(x) \rightarrow h(x)) \leftrightarrow \forall_x(f(x)\ \&\ \neg h(x) \rightarrow \neg g(x)))$. Здесь $f(x)$ и $h(x)$ не содержат неизвестных; $g(x)$ - содержит неизвестные и имеет своим заголовком равенство.

Попытка опровергнуь квантор общности

Если решается задача на описание, в которой требуется выявить истинность либо ложность условий, причем условием ее служит квантор общности, то предпринимается попытка установить ложность условия с помощью вспомогательной задачи на доказательство. Теорема приема имеет вид $\forall_{PQ}(\exists_x(P(x)\ \&\ \neg Q(x)) \rightarrow \neg(\forall_x(P(x) \rightarrow Q(x))))$. Фильтр "не(контекст(неизвестная(x1)))" означает, что задача не имеет неизвестных, т.е. решается для установления истинности либо ложности условий. Указатель "следствие(1)" определяет обращение к вспомогательной задаче на доказательство для проверки антецедента теоремы.

Нормализатор "нормдлялюбого"

Ввиду простоты применяемых здесь преобразований, ограничимся кратким их перечислением:

1. Если консеквент имеет вид конъюнкции, то кванторная импликация преобразуется к виду конъюнкции импликаций;
2. Если антецедент имеет вид $x = t$, где x - переменная кванторной приставки, не входящая в t, то этот антецедент отбрасывается; вместо всех оставшихся вхождений x подставляется t, и x исключается из кванторной приставки;
3. Если консеквент кванторной импликации совпадает с одним из ее антецедентов, то импликация заменяется на константу "истина";
4. Антецедент, представляющий собой логическую константу "истина", отбрасывается;
5. Если антецедент имеет вид дизъюнкции, то кванторная импликация приводится к виду конъюнкции импликаций;
6. Если консеквент и некоторый антецедент имеют заголовок "не", то эти отрицания отбрасываются, а консеквент и антецедент меняются местами;
7. Если консеквент имеет вид кванторной импликации, то предпринимается слияние внутреннего квантора общности с внешним. Антецеденты кванторных импликаций при этом объединяются.

3.9 Набор

В разделе собраны несколько простых приемов, относящихся к символам "слово", "префикс", "суффикс", "конкатенация", "слияние". Утверждение "слово(x)" означает, что x есть функция, определенная на начальном отрезке натурального ряда, т.е. понятие "слово" формализует понятие "упорядоченный набор". Так как неупорядоченные наборы нами не рассматриваются, мы для краткости повсюду будем употреблять в неформальных контекстах термины "слово"и "набор" как синонимы.

Понятие "слияние" представляет собой небольшую модификацию понятия "конкатенация". Если конец первого набора совпадает с началом второго, то при слиянии наборов повторного рассмотрения их общего граничного элемента не происходит. Такая операция удобна для работы с последовательно проходимыми путями в графах.

Среди представленных в разделе приемов упомянем прием, усматривающий истинность утверждения "функция(x)" для наборов x; прием, устраняющий вложенные конкатенации или слияния; прием, обращающийся к проверочному оператору "усмслово" для проверки истинности утверждения "слово(x)".

Наконец, отметим наличие проверочного оператора "усмслово", распознающего наборы - результаты применения операций "конкатенация", "префикс", и т.п.

3.10 Упражнения

Приведем ряд упражнений, в которых требуется реализовать на ГЕНОЛОГе различные общелогические приемы. В основном, эти приемы уже реализованы на ЛОСе, так что перед проверкой работоспособности созданного приема следует отключить старую его версию. Она легко находится через оглавление программ. Для отключения приема можно поместить в начало его программы какой-либо тождественно ложный оператор, например, оператор "равно(0 1)". Разумеется, после тестирования нового приема эту вставку следует удалить, а также удалить новый прием.

1. Создать прием, исключающий повторяющиеся операнды конъюнкции;

2. Создать прием, отбрасывающий такой член дизъюнкции, который получается из другого ее члена добавлением конъюнкции некоторых утверждений;

3. Создать прием, реализующий логическое упрощение $(A \mathbin{\&} B) \vee (A \mathbin{\&} \neg B) \to A$;

4. Создать прием, исключающий повторяющиеся антецеденты кванторной импликации;

5. Создать прием, устраняющий квантор существования в антецеденте кванторной импликации;

6. Создать прием, исключающий связанную переменную x кванторной импликации, если консеквент этой импликации имеет вид $\neg(x = t)$, где t не содержит x;

7. Создать пакетный нормализатор "нормэквивалентно" для приведения к стандартному виду эквивалентностей. Занести в него несколько приемов общей стандартизации эквивалентности, перечисленных в соответствующем разделе оглавления программ;

8. Создать прием, преобразующий выражение "префикс(a префикс(b c))" в "конкатенация(набор(a b)c)";

9. Создать прием, сводящий задачу на доказательство кванторной импликации с антецедентами $A_1, \ldots, A_n$ и консеквентом B к задаче на доказательство утверждения B из списка посылок, пополненного указанными антецедентами.

3.10.1 Указания

1. Входим в оглавление базы приемов; выбираем раздел "Логические приемы" - "Общие приемы" - "Конъюнкция", и вводим в нем новый концевой пункт. Входим в это пункт, нажимаем "Ctr-ф", и приступаем к вводу теоремы приема. Чтобы устранять повторяющиеся конъюнктивные члены, достаточно теоремы $\forall_x(x \; \& \; x \leftrightarrow x)$. При компиляции преобразований замены для ассоциативно-коммутативных операций создается программа, извлекающая часть необходимых для идентификации операндов и сохраняющая без изменения прочие операнды. Поэтому наш прием будет применим к конъюнкциям любого числа утверждений. После ввода теоремы под ней появляется горизонтальная линия. Нажимаем Enter, и вводим текстовым редактором заголовок приема - "второйтерм". Далее появляется третье окно. В нем размещаем фильтр "уровень(0)", определяющий уровень срабатывания приема. В случае приемов сканирования задачи указание в третьем окне уровня срабатывания является необходимым. Четвертое окно игнорируем - нажимаем "Esc". Под ним появляется голубая линия, означающая, что ввод приема завершен. Необходимо сразу же сохранить прием. Если нет уверенности в том, что прием уже готов к компиляции, лучше сохранить прием с помощью клавиши F4 - без компиляции. Это - обычная практика. Однако, в нашем случае прием крайне прост. Он не требует даже нормализаторов. Поэтому нажимаем сразу F3. Компиляция завершается успешно, о чем свидетельствует черная нижняя линия. Теперь можно посмотреть программу приема, нажав клавишу "Home". Входим в просмотр программы и нажимаем клавишу "курсор вверх". Она переводит к фрагменту программы, содержащему контрольную точку "прием(2)". Читая текст программы после контрольной точки, обнаруживаем, что здесь размещается старая версия приема, только что реализованного на ГЕНОЛОГе. Помещаем перед оператором "заменавхождения(...)" вставку "равно(0 1)", чтобы заблокировать работу старой версии. Теперь протестируем работу нового приема. Выйдем в задачник, выберем какой-либо его раздел, и создадим, например, задачу на описание с единственной неизвестной x и условием $(x = 0 \; \& \; x = 0) \; \vee \; x = 1$. Запустим процесс решения этой задачи клавишей "р". На первом же шаге сработает новый прием, заменяющий условие на $x = 0 \; \vee \; x = 1$. В заключение возвратимся к просмотру старой версии программы приема и уберем заглушку "равно(0 1)". Затем перейдем в базу приемов, удалим нажатием Ctr-Del новый прием, и удалим созданный для этого приема концевой пункт оглавления.

2. Теорема приема имеет вид $\forall_{ab}(a \; \vee \; (a \; \& \; b) \leftrightarrow a)$. При просмотре программы приема убеждаемся, что кроме выявления двух дизъюнктивных членов вида a, $a \; \& \; b$, она идентифицирует также случай, когда a - дизъюнкция нескольких членов, одновременно являющаяся конъюнктивным множителем текущего члена.

3. Теорема приема имеет вид $\forall_{AB}((A \,\&\, B) \lor (A \,\&\, \neg B) \leftrightarrow A)$.

4. Теорема приема имеет вид $\forall_{ABC}(\forall_x(A(x) \,\&\, A(x) \,\&\, B(x) \to C(x)) \leftrightarrow \forall_x(A(x) \,\&\, B(x) \to C(x)))$. Заметим, что антецеденты и консеквент необходимо представлять в виде функциональных переменных $A(x)$, $B(x)$, $C(x)$. Простые переменные A, B, C будут идентифицироваться как утверждения, не содержащие переменных связывающей приставки x. Заголовок приема - "второйтерм", единственный фильтр - "уровень(0)". Однако, в четвертом окне уже появляются элементы, необходимые для компиляции. Прежде всего, вводится указатель "кортежпеременных(x23)". Он означает, что теоремная переменная x будет идентифицироваться не с единственной переменной, а со связывающей приставкой произвольной длины. Так как выражения $A(x)$, $B(x)$, $C(x)$ нужно идентифицировать не с термами вида "значение(...)", а с произвольными термами, вводится указатель "отображение(x26 x27 x28)". Рекомендуется блокировать попытки компилятора рассматривать идентифицируемые кванторы с точностью до отрицания. Для этого вводим указатель "внешнийквантор(фикс(0 1))". Наконец, нужно объяснить компилятору, из каких соображений антецеденты разбиваются на три группы - два раза $A(x)$ и один раз $B(x)$. Нам нужно рассматривать $A(x)$ как произвольные единичные антецеденты. Поэтому вводим указатель "антецедент(x26 фикс(0 1))".

   Чтобы научиться аккуратно обращаться на ГЕНОЛОГе с функциональными переменными и кванторами, нужен некоторый опыт. Рекомендуется в затруднительных случаях находить аналогичные приемы. Иногда может понадобиться даже расширение ГЕНОЛОГа и развитие компилятора. Впрочем, приемы, требующие функциональных переменных, сравнительно редки.

5. Для исключения квантора существования, являющегося антецедентом кванторной импликации, можно использовать теорему вида: $\forall_{ABC}(\forall_x(\exists_y(A(x,y)) \,\&\, B(x) \to C(x)) \leftrightarrow \forall_{xy}(A(x,y) \,\&\, B(x) \to C(x)))$. Чтобы x, y идентифицировались со связывающими приставками произвольной длины, вводим указатели "кортежпеременных(x23)", "кортежпеременных(x24)". Функциональные переменные A, B, C выделяем указателем "отображение(x26 x27 x28)". Наконец, указатели "внешнийквантор(фикс(0 1))". "внешнийквантор(фикс(0 1 3))" добавляем, чтобы кванторы общности и существования идентифицировались без попыток перехода к их отрицаниям.

6. Если консеквент кванторной импликации имеет вид $\neg(x = t)$, где x - переменная связывающей приставки, а выражение t не содержит x, то можно перенести в антецеденты равенство $x = t$, а консеквентом сделать отрицание одного из антецедентов. После этого переменная x легко устраняется. Теорему приема, выполняющего данные преобразования, можно записать в следующем виде: $\forall_{ABt}(\forall_{xy}(A(x,y) \,\&\, B(x,y) \to \neg(x = t(y))) \leftrightarrow \forall_y(A(t(y),y) \to \neg B(t(y),y)))$. Чтобы компилятор идентифицировал $B(x,y)$ как произвольный антецедент импликации, вводим указатель "антецедент(x27 фикс(0 1))". Если желательно выбрать какой-либо специальный антецедент, то достаточно ввести фильтр, отбирающий такой антецедент. Например, чтобы, по возможности, исключить появление отрицания в консеквенте, будем при наличии хотя бы одного отрицания в антецедентах брать в качестве $B(x,y)$ только антецедент с отрицанием. Соответствующий фильтр имеет вид "или(заголовок(фикс(0 1

5)не)не(контекст(антецедент(фикс(0 1)x4)заголовок(x4 не))))". Указатель "кортежпеременных(x24)" позволяет рассматривать произвольные списки y связанных переменных, дополняющие x. Указатель "внешнийквантор(фикс(0 1))" сопровождает идентифицируемый квантор общности; указатель "отображение(x19 x26 x27)" выделяет функциональные переменные A, B, t. Чтобы убрать возможное двойное отрицание в утверждении $\neg B(t(y), y)$, сопровождаем его нормализатором "нормлог". Заметим, что прием будет обрабатывать и вырожденные случаи - отсутствие антецедентов $A(x, y)$, отличных от $B(x, y)$, а также случай пустого списка y. Вместо кванторной импликации в последнем случае будет создана дизъюнкция.

7. Чтобы создать новый нормализатор "нормэквивалентно", используем вспомогательный интерфейс. Нажимаем клавишу Ctr-n. Так как нам нужен нормализатор общей стандартизации, название его можно пока не вводить. Оно будет создано автоматически. Достаточно выбрать окно "Нормализатор" и нажать левую кнопку мыши. Далее нажимаем левую кнопку мыши в окне "Нормализатор общей станартизации …". Появляется курсор текстового редактора. Вводим название того логического символа, для которого предназначен нормализатор. В нашем случае это символ "эквивалентно". После нажатия Enter, завершающего ввод символа, появляются также число уровней срабатывания (по умолчанию 3) и указатели на наличие списка посылок и корневой режим обработки терма. Нажимаем Enter и возвращаемся в оглавление базы приемов, где обнаруживаем подраздел для нового нормализатора "нормэквивалентно". Здесь уже имеются приемы справочников, характеризующих нормализатор, а также корневой фрагмент программы нормализатора (пункт "Переключатель уровня").

   Рассмотрим первые два простейших приема нормализатора, предоставляя читателю продолжить его развитие самостоятельно. Эти приемы имеют теоремы $\forall_x((x \leftrightarrow x) \leftrightarrow \text{истина})$, $\forall_x((x \leftrightarrow \neg x) \leftrightarrow \text{ложь})$. Заметим, что консеквент теоремы приема имеет вид эквивалентности и заменяемая часть тоже имеет вид эквивалентности. Поэтому заменяемую часть необходимо заключить в скобки (иначе проблемы появятся уже на этапе работы с формульным редактором). Кроме того, заменяюшая часть указанных двух теорем представляет собой логическую константу, которую следует указать явно ("истина" вводится двойным нажатием "t", "ложь" - двойным нажатием "f". Если бы в первой теореме логическая константа отсутствовала, а консеквент имел вид $x \leftrightarrow x$), то компилятор создал бы программу, заменяющую текущий преобразуемый терм на самого себя.

8. Теорема приема имеет вид $\forall_{abc}(\text{префикс}(a, \text{префикс}(b, c)) = ((a, b); c))$. Точка с запятой обозначает здесь конкатенацию наборов. Хотя после ввода текста теоремы пара (a, b) будет перерисована без скобок, ее нужно вводить заключенной в скобки. Если не вводить никаких указателей, то прием откомпилирован не будет. Причиной этого послужит невозможность выбора в теореме точки привязки. Ведь по умолчанию выражение "префикс(A B)" идентифицируется с задающим набор термом достаточно произвольного вида - либо "набор(…)", либо "префикс(…)", либо "конкатенация(…)". Нет никакого логического символа, который обязательно встречался бы в каждом из указанных случаев и который можно было бы выбрать в качестве символа привязки. Поэтому вводим указатели "нормнабор(фикс(0 1))" и "нормнабор(фикс(0 1 2))", означаю-

щие, что символ "префикс" идентифицируется буквально. С ними компиляция приема становится возможной.

9. В этом примере мы встречаемся с ситуацией, когда перед выполнением преобразования необходимо выполнить проверку некоторого утверждения B относительно заданных посылок A. Ее можно было бы организовать, снабдив теорему приема антецедентом типа $\forall_x(A(x) \to B(x))$ и обратившись к вспомогательной задаче на его доказательство. Однако, удобнее применять другой способ - ввести в антецеденты теоремы только $B(x)$, а посылки $A(x)$ передавать вспомогательной задаче через специальные указатели. Тогда теорема приема, без учета сопровождающих указателей, приобретает несколько странный вид: $\forall_{AB}(B(x) \to \forall_x(A(x) \to B(x)))$. Вводим заголовок приема "второйтерм", а также уточняющие контекст фильтры: "уровень(3)"; "тип(доказать)"; "условие"; "корень". Для обработки антецедента вводим указатель "следствие(1)", определяющий обращение к вспомогательной задаче на доказательство. Максимальный уровень обращения совпадает с максимальным уровнем текущей задачи. Чтобы передать этой задаче в качестве дополнительных посылок конъюнкцию антецедентов $A(x)$, используем указатель "занесениепосылки(1 значение(x26 x23))". Добавляем стандартные указатели "кортежпеременных(x23)", "отображение(x26 x27)"и "внешнийквантор(фикс(0))".

Глава 4

Приемы по алгебре множеств

В последующем изложении мы постараемся упорядочить разделы базы приемов в естественном порядке. Однако, во многих случаях приемы, отнесенные к какому-либо разделу, будут использовать аппарат последующих разделов. Так как логический язык системы уже был описан в первой книге, то принципиальных затруднений эти случаи вызвать не должны. Более подробно разбираются те приемы, которые иллюстрируют какие-либо особенности программирования на ГЕНОЛОГе. Иногда достаточно будет лишь краткого перечисления имеющихся в разделе приемов и общей характеризации логики управления ими. Никакого создания приемов "из общих соображений" не предпринималось. Таким образом, их перечень интересен в двух отношениях: как эмпирически выявленный "базис" средств, необходимых для решения задач (видимо, пока далеко не полный), и как иллюстрация принципов программирования решателей на ГЕНОЛОГе. Рекомендуется в качестве дополнительного упражнения находить описываемые ниже приемы в базе решателя и определять те задачи, в которых они применяются.

В данной главе рассматриваются приемы, относящиеся к простейшим теоретико-множественным предикатам и операциям. К их числу относятся предикаты "множество(x)", "принадлежит(x y)", "содержится(x y)", "непересек(x y)", символ пустого множества "пусто" и операции "объединение", "пересечение", "разность", "симметричразность", "прямоепроизведение". Здесь же рассматриваются приемы для конечных множеств, задаваемых перечислением своих элементов, а также приемы для предикатов и операций над семействами множеств. Как правило, приемы раздела основаны на очень простых теоремах и имеют столь же простое управление. Достаточно сказать, что в процессе развития системы неоднократно предпринимались сравнительно удачные попытки автоматического воссоздания всего кластера приемов по алгебре множеств из исходных определений. В то же время, при переходе к автоматическому синтезу приемов по элементарной алгебре возникли настолько серьезные трудности, что работа в этом направлении на долгий период была приостановлена. Многие из представленных в разделе приемов были первоначально сгенерированы автоматически, а при последующих обновлениях версий системы сохранены. Этим объясняются имеющееся во многих случаях "обратное" направление применения тождества либо эквивалентности, а также явная перенасыщенность ряда разделов группами простых тождеств.

Войдя в раздел оглавления базы приемов "Теория множеств", прежде всего обратим внимание на первый его подраздел - "Справочники". В таких подразделах группируются общие приемы справочников, обслуживающих понятия предметной области. Выбираем пункт "Содержание" и обнаруживаем теорему приема "содержание(теориямножеств множество принадлежит содержится ...)". Здесь перечисляются все логические символы, отнесенные к разделу "теориямножеств", а также названия всех его подразделов. Символы упорядочены по возрастанию их эвристической "сложности", и это упорядочение является эталонным для ряда процедур решателя, где необходим выбор "самого сложного" понятия в заданном терме. Главным образом, оно используется для равномерного распределения теорем по понятиям. Справочник "содержание" позволяет получать список понятий и названий подразделов по названию раздела.

Справочник "раздел" дает встречную ссылку - по понятию указывает на содержащий его раздел. Всякий раз, как в логическую систему вводится новое понятие, оно регистрируется в каком-либо разделе через справочники "содержание" и "раздел".

4.1 Общие приемы для множеств

Рассмотрим приемы, собранные в подразделе "Множество" раздела "Теория множеств". Прежде всего, отметим справочник "родобъекта", указывающий, что "множество" - один из основных типов объектов. Основные типы объектов образуют частично упорядоченное множество. Если два типа несравнимы (ни один не является подтипом другого), то объекты этих типов обязательно различны. Например, никакое множество не является функцией или числом. Чтобы определить по данному основному типу все его надтипы, используется справочник "род".

Усмотрение множества

Утверждения вида "множество($f(\ldots)$)", где f - некоторая операция, часто бывают избыточными из-за того, что любые значения операции f суть множества. Чтобы убедиться в этом, используется справочник "тип", дающий для f список всех основных типов, к которым относится каждое значение f. Считается, что это выполнено, даже если операция применяется к находящемуся вне ее о.д.з. набору операндов. Прием, выполняющий обращение к справочнику "тип" и замену утверждений указанного вида на константу "истина", основан на теореме "родобъекта(множество)". Его заголовок - "родобъекта".

Еще один прием, обеспечивающий усмотрение множества, основан на теореме $\forall_{af}(a = \mathrm{set}_x(f(x)) \to a - \mathrm{set})$. Он заменяет на константу "истина" утверждения "множество(a)", для которых в контексте имеется равенство, определяющее a через описатель "класс".

Наконец, для усмотрения множества в несколько более сложных ситуациях используется проверочный оператор "усммножество". Теорема приема, обращающегося к нему, имеет вид $\forall_a(a - \mathrm{set} \to a - \mathrm{set})$. Антецедент сопровожден указателем "блокпроверок(1)", и компилятор преобразует его в обращение к оператору "усммножество". Важно отметить, что устранение условий либо посылок задач, имеющих вид "множество(x)", где x - переменная, может привести к полному нарушению нормальной работы решателя, даже если истинность таких утверждений усматривается из их контекста. Они характеризуют типы значений переменных задачи и используются во многих приемах. Это же относится и к другим основным типам, таким, как "число", "функция", и т.п. Поэтому в приеме, обращающемся к оператору "усммножество", предусмотрен фильтр "или(не(корень)не(переменная(x1))и(тип(описать)цель(исследовать)))".

Усмотрение противоречивых указаний на тип объекта

Если в задаче встретилось утверждение "множество(A)", то проверяется наличие в его контексте утверждения $P(A)$, где P - основной тип объекта, несравнимый с типом "множество". Если таковое обнаружено, то утверждение "множество(A)" заменяется на константу "ложь". Прием, выполняющий данную проверку, имеет теорему "родобъекта(множество)" и заголовок "различимы".

Еще один прием связан с утверждениями вида "множество($f(\ldots)$)", где список основных типов значений операции f не содержит символа "множество". Эти утверждения тоже заменяются на константу "ложь". Теорема приема - та же, что и выше; заголовок приема - "смзначение".

Подбор примера

Если в задаче на описание, решаемой с целью "пример", имеется едиственное содержащее неизвестную x условие "множество(x)", то предпринимается попытка выбрать в качестве значения x пустое множество. Теорема приема - $\forall_a(a = \emptyset \rightarrow a - \mathrm{set})$. Заголовок приема "подборзначений" и указатель "подборзначений(1)" означают, что будет решаться вспомогательная задача, полученная из текущей заменой условия $a - \mathrm{set}$ на $a = \emptyset$. Уровень срабатывания приема равен 1.

Важная деталь - наличие в приеме указателя "новый". Он означает, что в начальной части программы приема оператор "новый" отсутствует. Таким образом, при каждом выходе утверждения $a - \mathrm{set}$ из теневой зоны сканирования будут предприниматься повторные попытки применения данного приема, даже если вес рассматриваемого утверждения окажется больше 1. Это устраняет эффект "слепого пятна", когда решатель не замечает уже найденного ответа из-за того, что при первых попытках применения приема еще имелись другие условия, содержащие неизвестную a, а когда они исчезли, вес условия "$a - \mathrm{set}$" стал большим.

Кванторная расшифровка условия равенства множеств

Преобразование условия равенства двух множеств в кванторную эквивалентность условий принадлежности этим множествам используется только при доказательстве равенства множеств. Более того, это делается лишь на сравнительно высоком уровне - если не срабатывают приемы, позволяющие установить равенство более простыми средствами (например, с помощью тождественных преобразований). Теорема приема имеет вид $\forall_{ab}(a - \mathrm{set} \;\&\; b - \mathrm{set} \rightarrow (a = b \leftrightarrow \forall_c(c \in a \leftrightarrow c \in b)))$. Заметим, что антецеденты, уточняющие тип значений переменных a, b, здесь необходимы - без них невозможно было бы определить, что речь идет о множествах. С другой стороны, аналогичные антецеденты в приемах, где переменные для множеств расположены в идентифицируемой части теоремы под теоретико-множественными операциями и отношениями, можно опускать. Там они являются следствием стандартных требований на о.д.з. и, ввиду наличия системы сопровождения о.д.з., избыточны. Прием применяется, если условие задачи на доказательство либо совпадает с указанным равенством, либо является его отрицанием. В первом случае уровень срабатывания равен 6, во втором - 7. Большее значение уровня срабатывания в случае отрицания объясняется тем, что тогда квантор общности преобразуется в квантор существования, и далее приходится обращаться к задаче на описание. Если же отрицания нет, то квантор общности в условии задачи на доказательство будет просто отброшен.

Чтобы блокировать обратное преобразование, задача сопровождается комментарием "кванторнаясвертка".

Кванторная свертка в условие равенства множеств

Кванторная эквивалентность двух условий принадлежности множествам обычно сразу же (на уровне 0) преобразуется в условие равенства этих множеств. Это преобразование блокируется комментарием "кванторнаясвертка". Теорема приема имеет вид $\forall_{ab}(a = b \leftrightarrow \forall_c(c \in a \leftrightarrow c \in b))$; заголовок приема - "первыйтерм". Заметим, что антецеденты, уточняющие тип значений переменных a, b, при такой замене не требуются: в задаче имеются условия принадлежности, из которых ясно, что при соблюдении соглашений относительно о.д.з. объекты a и b должны быть множествами. Слабые ограничения на применение приема связаны, во-первых, с редактированием параметрических описаний, и, во-вторых, с процедурами вывода теорем. Указатель "кванторнаясвертка" означает, что попытка применить прием будет выполняться не только для квантора общности, но и для квантора существования (виртуально представляемого как отрицание квантора общности). При этом переменная c может быть в кванторной приставке не единственной - прочие переменные и не содержащие c подкванторные утверждения будут при идентификации вынесены за скобку.

Параметрическое описание для отрицания равенства двух множеств

Если отрицание равенства двух неизвестных множеств - условие задачи на описание, в которой требуется получить пример либо параметрическое описание, то на достаточно высоком уровне (6-м) оно может быть заменено на условие существования элемента, принадлежащего одному множеству и не принадлежащего другому.

Исключение квантора существования

Условия существования множества либо непустого множества заменяются на константу "истина".

Попытка представить уравнение для множества в виде двух включений

Уравнения с множествами чаще всего решаются путем рассмотрения двух встречных включений. Прием, выполняющий такую замену, основан на теореме $\forall_{ab}(a - \mathrm{set}\ \&\ b - \mathrm{set} \rightarrow (a = b \leftrightarrow a \subseteq b\ \&\ b \subseteq a))$. Однако, если просто выполнить замену согласно теореме, то сразу же сработал бы обратный прием общей стандартизации, преобразующий пару встречных включений в равенство множеств. Поэтому данный прием в действительности делает другое. Он обращается к нормализаторам "нормусм" и "уравнсодержится", первый из которых предпринимает попытку усмотреть истинность каждого из включений, а второй - разрешить его относительно неизвестных. Если после этого оказывается, что глубина вхождений неизвестных в результирующие утверждения уменьшилась по сравнению с глубиной их вхождений в исходное равенство, то замена реализуется. В этом случае включения будут изменены, и срабатывания обратного приема не произойдет.

Проверочный оператор "усммножество"

Для усмотрения истинности утверждений "множество(A)" введен проверочный оператор "усммножество". Найдем в базе приемов раздел, соответствующий этому оператору. Прежде всего, обратим внимание на подраздел "Справочники". В пункте "Легковидеть" представлен прием одноименного справочника, определяющий формат оператора. Теорема этого приема - "легковидеть(усммножество(x1 x2 x3 x4)множество(x1))". В ней указаны шаблон "множество(x1)" подлежащих обработке оператором утверждений, а также шаблон обращения к оператору. Указатель "уровень(...)" отсутствует, т.е. число уровней срабатывания проверочного оператора равно 1. Справочник "очевидно" основан на той же теореме. Он позволяет найти по заголовку проверочного оператора вид проверяемых утверждений. Заметим, что отсутствие справочника "очевидно" негативно сказывается на качестве трассировки "по шагам решения". В этом случае каждая попытка обращения к проверочному оператору вызывает ненужные точки прерывания, затрудняющие восприятие основных действий. Поэтому, вводя новый проверочный опетатор, не следует забывать сопровождать его приемом справочника "очевидно".

После подраздела "Справочники" идут два пункта, имеющиеся у любых проверочных операторов. Первый из них - "Усмотрение результата из буфера" - содержит прием "контрольбуфера". Он предпринимает попытку извлечь готовый результат из буфера ранее выполнявшихся обращений к оператору. Второй пункт - "Непосредственное усмотрение" - содержит прием "быстрпроверка", обращающийся к процедуре "стандследствие" для усмотрения истинности утверждения из общих соображений. Например, проверяется наличие данного утверждения в списке посылок. Теорема обоих приемов - та же, что в справочнике "легковидеть".

Далее начинаются собственные приемы проверочного оператора "усммножество":

1. В посылках имеется равенство $a = b$. Тогда, чтобы проверить утверждение "множество(b)", проверяется утверждение "множество(a)". Никаких специальных указателей приема, предотвращающих зацикливание, здесь не требуется. При рекурсивном обращении к проверке утверждения "множество(b)" автоматически вводится комментарий (исключение $(a = b)$), блокирующий повторное использование равенства. По умолчанию, в таких комментариях регистрируются все посылки, непосредственно идентифицированные с антецедентами теорем приемов проверочных операторов.

2. В посылках имеется утверждение "семействомножеств($\lambda_x(f(x), g(x))$)". Тогда, чтобы проверить утверждение "множество($f(a)$)", предпринимается обращение к вспомогательной задаче на доказательство утверждения $g(a)$. Введено достаточно жесткое ограничение на трудоемкость этой попытки.

3. В посылках имеется равенство $A = \mathrm{set}_x(P(x))$. Тогда утверждение "множество(A)" усматривается непосредственно.

4. В посылках имеется утверждение "семействомножеств(A)". Утверждение "множество($A(i)$)" усматривается непосредственно.

5. Если нужно усмотреть утверждение "множество((A при P, иначе B))", то выполняются проверки утверждений "множество(A)", "множество(B)".

Нормализатор уравнений "уравнмножество"

Это - один из четырех пакетных нормализаторов, используемых для разрешения относительно неизвестных простейших теоретико - множественных соотношений. Он относится к равенствам множеств; остальные три - к соотношениям принадлежности, включения и непересечения. Большинство приемов данных нормализаторов продублированы в виде приемов сканирования задачи. Обычно прием сканирования задачи делает первый шаг цепочки эквивалентных преобразований, усматривая некоторое соотношение с неизвестными, а остальные шаги реализуются внутри нормализатора. Это позволяет получить определенное ускорение вычислений. Специальных приемов сканирования задачи, обращающихся к указанным нормализаторам без каких-либо собственных преобразований условия, не предусмотрено.

Нормализатор обрабатывает единственное уравнение, причем ему передается в комментариях список неизвестных. Это позволяет использовать фильтры "известно(...)", "неизвестная(...)" так же, как в приемах сканирования задачи. Предусмотрены следующие приемы:

1. Утверждения $a = a \cap b$, $a = a \cup b$, $a = a \setminus b$ преобразуются, соответственно, в $a \subseteq b$, $b \subseteq a$, непересек(a, b). Каждое из последних обрабатывается нормализатором "нормусм". Этот нормализатор предназначен для усмотрения заведомо истинных либо ложных утверждений. Он определяет, существует ли проверочный оператор, который мог бы распознать истинность обрабатываемого утверждения U либо его отрицания. Если такой оператор находится, то реализуется обращение к нему. В случае успеха утверждение U заменяется на соответствующую логическюу константу. Нормализатор "нормусм" имеет достаточно универсальных характер и часто используется в других разделах (например, в элементарной алгебре).

2. Утверждение $a \cap b = \emptyset$ преобразуется в утверждение "непересек(a,b)", к которому применен нормализатор "нормусм";

3. Утверждение $a \setminus b = \emptyset$ преобразуется в утверждение $a \subseteq b$. Последнее обрабатывается нормализаторами "нормусм" и "уравнсодержится;

4. Утверждение "образ(b a) = $\emptyset$" преобразуется в утверждение $a = \emptyset$, к которому применен нормализатор "нормусм";

5. Утверждение $a \times b = \emptyset$ заменяется на дизъюнкцию $a = \emptyset \ \vee \ b = \emptyset$. К каждому из дизъюнктивных членов применены нормализаторы "нормусм" и "уравнмножество", а ко всей дизъюнкции - нормализатор "нормлог";

6. Утверждение $a \cup b = \emptyset$ заменяется на конъюнкцию $a = \emptyset \ \& \ b = \emptyset$. Нормализаторы - такие же, как в предыдущем случае;

7. Утверждение $a \times b = c \times d$ заменяется на дизъюнкцию $((a = \emptyset \ \vee \ b = \emptyset) \ \& \ (c = \emptyset \ \vee \ d = \emptyset)) \ \vee \ (a = c \ \& \ b = d)$;

8. Если преобразуемое утверждение имеет вид $c = \emptyset$, где выражение c не является переменной, не имеет заголовка "объединение" и содержит неизвестные, то находится результат a преобразования выражения c к виду объединения пересечений либо разностей. Для это используется нормализатор "видобъединение", выполняющий всю необходимую работу по "раскрыванию скобок" применительно к теоретико - множественным операциям. Если результат a имеет своим заголовком объединение либо длина его меньше длины c, то исходное утверждение заменяется на утверждение $a = \emptyset$. Последнее обрабатывается нормализаторами "нормусм" и "уравнмножество". Фильтр "Длина(Контрольглубины)" обеспечивает проверку уменьшения глубины вхождений неизвестных.

9. Если преобразуемое утверждение содержит неизвестные и имеет вид $e = c \times d$, где выражение e отлично от переменной и не имеет заголовка "прямоепроизведение", то предпринимается попытка преобразовать e к виду прямого произведения. Для этого используется нормализатор "видпрямоепроизведение". Если результат преобразования a имеет своим заголовком прямое произведение либо длина его меньше длины e, то исходное утверждение заменяется на $a = c \times d$. Как и в предыдущем случае, далее применяется нормализатор "уравнмножество" и происходит контроль уменьшения глубины вхождений неизвестных.

Нормализатор "группмножество"

Обычно приемы нормализаторов сразу принимают решение о целесообразности текущего преобразования и реализуют это преобразование. Однако, в очень редких случаях может понадобиться другой режим работы нормализатора, состоящий в определении всего спектра альтернативных текущих преобразований и последующем выборе группы непересекающихся преобразований, дающей наилучший результат. В ГЕНОЛОГе была предусмотрена возможность создания таких нормализаторов. Они получили название нормализаторов группировки, так как первоначально предназначались для упрощения многочленных выражений путем группировок. Однако, при развитии системы остался востребованным лишь один нормализатор группировки - "группмножество". Это объясняется тем, режим группировок с последующими перегруппировками, реализуемый в обычных нормализаторах, оказался более эффективным, чем указанный выше режим полного просмотра текущих преобразований. В сущности, и для множеств нормализатор был оставлен в системе лишь как пример невостребованной, хотя и первоначально представлявшейся естественной, возможности.

Для создания нового нормализатора группировки достаточно включить в задающий его формат терм "быстрпреобр(...)" элемент "группировка". Приемы нормализатора будут регистрировать в накопителе варианты текущих преобразований с их оценками. По завершении полного просмотра приемов произойдет обращение к процедуре "замены(...)", которая отберет подмножество непересекающихся преобразований с наибольшей суммарной оценкой. Далее цикл повторится - до тех пор, пока накопитель не окажется пустым.

Приведем перечень теорем приемов нормализатора "группмножество". Во всех этих преобразованиях заменяющей является левая часть тождества. Ко всем операциям заменяющей части применяются соответствующие нормализаторы сокращенной перезаписи - "упрощобъединение", "упрощразность", и т.п.

1. $\forall_{abc}(\text{прообраз}(a, c \setminus b) = \text{прообраз}(a, c) \setminus \text{прообраз}(a, b))$;
2. $\forall_{abc}(\text{прообраз}(a, c \cap b) = \text{прообраз}(a, c) \cap \text{прообраз}(a, b))$;
3. $\forall_{abc}(\text{прообраз}(a, c \cup b) = \text{прообраз}(a, c) \cup \text{прообраз}(a, b))$;
4. $\forall_{abc}(\text{образ}(a, c \cup b) = \text{образ}(a, c) \cup \text{образ}(a, b))$;
5. $\forall_{abc}(a \times c \setminus b) = (a \times c) \setminus (a \times b)$;
6. $\forall_{abc}(c \setminus b \times a) = (c \times a) \setminus (b \times a)$;
7. $\forall_{abc}((a \cup b) \times c) = (a \times c) \cup (b \times c)$;

8. $\forall_{abc}(b \times (a \cup c) = (b \times a) \cup (b \times c)$;
9. $\forall_{abc}(c \cap (b \setminus a) = (b \cap c) \setminus a)$;
10. $\forall_{abc}((b \setminus c) \setminus a = b \setminus (a \cup c))$;
11. $\forall_{abcd}(a \setminus (b \setminus (c \setminus d)) = a \setminus b \cup (a \cap c) \setminus d)$;
12. $\forall_{abc}(c \setminus (a \cap b) = (c \setminus a) \cup (c \setminus b))$;
13. $\forall_{abcde}(e \cap (b \setminus a \cup d \setminus c) = ((b \cap e) \setminus a) \cup ((d \cap e) \setminus c))$;
14. $\forall_{abcde}((a \setminus b \cup d \setminus e) \setminus c = (a \setminus (b \cup c) \cup d \setminus (c \cup e)))$;

4.2 Приемы символа "пустое множество"

При описании приемов ГЕНОЛОГа, как и в случае приемов ЛОСа, будем относить их к каким-либо характерным логическим символам. Однако, в отличие от ЛОСа, где логический символ представлял собой точку обращения к приему при сканировании задачи, здесь выбор символа будет соответствовать лишь названию раздела оглавления базы приемов. Обычно он представляет собой чистую условность и не совпадает ни с символом обращения к программе приема, ни с символом, используемым для ссылок на прием в технических структурах ГЕНОЛОГа.

Подбор примера

Пустое множество часто используется в задачах, имеющих цель "пример". Если для некоторой неизвестной x, помимо условия "множество(x)", остается лишь единственное условие "$x \subseteq a$", либо "непересек$(x\ a)$", либо "$\neg(a \in x)$", то в качестве значения неизвестной берется $\emptyset$. Приемы для этих трех ситуаций однотипны. Например, в первом случае прием имеет теорему: $\forall_{ax}(x = \emptyset \rightarrow x \subseteq a)$. Заголовок приема - "подборзначений". Указатели и фильтры приема аналогичны уже встречавшимся ранее - см. подбор примера для условия "множество(x)".

Усмотрение соотношений с пустым множеством

Введены приемы, усматривающие истинность соотношений "$\neg(x \in \emptyset)$", "непересек$(x, \emptyset)$", "$\emptyset \subseteq x$".

Включение в пустое множество

Соотношение включения в пустое множество заменяется на равенство этому множеству.

Кванторная свертка в условие пустоты множества

Утверждение вида $\forall_x(\neg(x \in a))$ обычно преобразуется к виду $a = \emptyset$. Исключение составляют лишь несколько редко встречающихся случаев. Предусмотрена идентификация квантора с попыткой перехода к отрицанию. Таким образом, утверждение $\exists_x(x \in a)$ прием будет заменять на $\neg(a = \emptyset)$.

Условное выражение с пустым множеством

Если одним из альтернативных выражений условного выражения a является символ пустого множества, то предпринимается попытка усмотреть непустоту значения a. Теорема приема имеет вид $\forall_{abc}(\neg(a=\emptyset)\to((a=(b \text{ при } c, \text{иначе } \emptyset))\leftrightarrow(c\ \&\ (a=b))))$.

Проверочный оператор "усмнепусто"

Чтобы усматривать непустоту множества, введен проверочный оператор "усмнепусто". В нем, как и во многих других проверочных операторах, достаточно часто используются рекурсивные обращения. Чтобы избежать зацикливания при таких обращениях, имеется два способа. Во-первых, обычно условие, проверяемое в антецеденте теоремы приема, проще условия в ее консеквенте. Во-вторых, если антецедент теоремы приема идентифицируется непосредственно с посылкой, то эта посылка запоминается в комментариях, и при рекурсивном обращении к проверке другого антецедента уже не используется. Впрочем, последнее ограничение в особых случаях может отменяться.

Перечислим некоторые из приемов оператора "усмнепусто":

1. Усмотрение непустоты множества из включения в него непустого множества. Теорема имеет вид $\forall_{ab}(a - \text{set}\ \&\ a\subseteq b\ \&\ \neg(a=\emptyset)\to\neg(b=\emptyset))$. Включение усматривается в посылках явным образом, а первый и третий антецеденты обрабатываются проверочными операторами.

2. Усмотрение непустоты множества из невключения его в другое множество. Теорема имеет вид $\forall_{ab}(\neg(b\subseteq a)\to\neg(b=\emptyset))$. Отрицание включения усматривается в посылках явным образом.

3. Усмотрение непустоты множества из пересечения его с другим множеством. Теорема имеет вид: $\forall_{ab}(\neg(\text{непересек}(a,b))\to\neg(b=\emptyset))$. Отрицание непересечения в явном виде содержится в посылках.

4. Усмотрение непустоты объединения двух множеств из непустоты одного из них. Теорема имеет вид $\forall_{ab}(a - \text{set}\ \&\ \neg(a=\emptyset)\to\neg(a\cup b=\emptyset))$. Оба антецедента обрабатываются проверочными операторами. Компилятор последовательно идентифицирует a с операндами объединения.

5. Непустота пересечения пересекающихся множеств. Теорема приема имеет вид $\forall_{ab}(\neg(\text{непересек}(a,b))\to\neg(a\cap b=\emptyset))$. Антецедент обрабатывается проверочным оператором. Заметим, что каждое из его входных данных a, b проще исходного входного данного $a\cap b$. Чтобы оператор "усмненепересек" или какой-либо другой проверочный оператор, возникающий в цепочке обращений, не создал зацикливания, обратившись к проверке непустоты пересечения $a\cap b$, прием снабжен "защелкой". Его применение блокируется при наличии комментария "перестановка", а комментарий этот передается оператору "усмненепересек". Напомним, что при рекурсивных обращениях между пакетными операторами старый список комментариев обычно лишь пополняется, так что комментарий передается по всей глубине цепочки обращений.

6. Непустота разности невключающихся друг в друга множеств. Теорема приема имеет вид $\forall_{ab}(\neg(a\subseteq b)\to\neg(a\setminus b=\emptyset))$. Антецедент обрабатывается проверочным оператором.

7. Непустота перечня. Теорема приема имеет вид $\forall_{ab}(\neg(\{b;a\}=\emptyset))$. Его программа усматривает терм "перечень(t)", убеждается, что можно определить первый элемент b набора t, и выдает результат.

8. Непустота прямого произведения непустых множеств. Теорема имеет вид $\forall_{ab}(\neg(b=\emptyset)\ \&\ a-\mathrm{set}\ \&\ b-\mathrm{set}\ \&\ \neg(a=\emptyset)\to\neg(a\times b=\emptyset))$. Все антецеденты обрабатываются проверочными операторами. Указатель "спуск" означает, что если попытка применить прием оканчивается неудачей, то другие приемы не рассматриваются - сразу выдается "отказ".

9. Усмотрение непустоты множества, мощность которого не равна нулю. Теорема имеет вид $\forall_a(\neg(\mathrm{card}a=0)\to\neg(a=\emptyset))$. Антецедент обрабатывается проверочным оператором "усмне0", причем предварительно предпринимается попытка вычислить мощность с помощью нормализатора "нормощность". Фильтр "контекст(посылка(x2)входит(мощность x2))" указывает, что в посылках должно иметься хоть какое-то упоминание о мощности множеств. Фильтр "коммент(усмне0)" предотвращает зацикливание: оператор "усмне0" делает обратное обращение к проверке непустоты, сопровождая ее комментарием "усмне0".

10. Усмотрение непустоты множества, заданного параметрически. Теорема имеет вид $\forall_{fg}(\neg(\mathrm{set}_y(g(y))=\emptyset)\to\neg(\mathrm{set}_x(\exists_y(x=f(y)\ \&\ g(y)))=\emptyset))$. Антецедент обрабатывается тем же самым проверочным оператором. Его подвыражение $\mathrm{set}_y(g(y))$ предварительно преобразуется нормализатором "нормкласс".

11. Непустота множеств целых и натуральных чисел, а также числового промежутка с различными концами. В последнем случае теорема имеет вид $\forall_{abcd}(0<b-a\to\neg([a,b]=\emptyset))$. Напомним, что промежуток $[a,b]$ во внутреннем представлении записывается с помощью четырехместного символа: "промежуток(a b c d)". Здесь c,d - указатели принадлежности концов промежутку. Если указатель равен 0, то конец отбрасывается; если он равен 1, то включается. Если в качестве указателя используется явное значение 0 либо 1, то формульный редактор будет прорисовывать, соответственно, круглую либо квадратную скобку. Если же роль указателя играет какое-либо иное выражение (например, переменная), то прорисовывается квадратная скобка. Это надо помнить при чтении теорем приемов и контролировать смысл квадратной скобки переходом от формульного к текстовому режиму просмотра. В нашем случае роль указателей играют переменные, т.е. прием относится к промежуткам произвольных типов.

12. Непустота множества, содержащего элемент. Теорема приема имеет вид $\forall_{ab}(b\in a\to\neg(a=\emptyset))$. Ее антецедент должен быть усмотрен в посылках непосредственно.

13. Непустота класса. Если множество задано с помощью описателя "класс", то для установления его непустоты решается вспомогательная задача на доказательство существования объекта, удовлетворяющего условию принадлежности классу. Теорема приема имеет вид $\forall_{fa}(a=\mathrm{set}_x(f(x))\ \&\ \exists_x(f(x))\to\neg(a=\emptyset))$. Первый антецедент извлекается из посылок. Для обработки второго введен указатель "легковидеть(2)", определяющий обращение к вспомогательной задаче. Она решается с существенным ограничением средств - максимальный уровень ее равен 4.

14. Непустота прообраза. Здесь представлено два приема: $\forall_{fAB}(\neg(\text{прообраз}(f, A) = \emptyset) \to \neg(\text{прообраз}(f, A \cup B) = \emptyset))$; $\forall_{AB}(B \subseteq A \,\&\, \neg(B = \emptyset) \,\&\, \text{перестановка}(f, A) \to \neg(\text{прообраз}(f, B) = \emptyset))$. Первый для установления непустоты прообраза объединения множеств пытается усмотреть непустоту прообраза одного из этих множеств; второй - использует взаимную однозначность перестановки. Антецедент "перестановка(f A)" извлекается из посылок в явном виде; прочие - обрабатываются проверочными операторами.

15. Непустота сомножителя непустого прямого произведения. Теорема имеет вид $\forall_{ABa}(\text{card}(A \times B) = a \;\&\; 0 < a \to \neg(A = \emptyset))$. Первый антецедент содержится в посылках, второй - проверяется. Аналогичная теорема введена для непустоты сомножителя B.

4.3 Приемы символа "принадлежит"

Общая характеризация символа обеспечивается справочниками "предикатныйсимвол", "одз", "типданных". Первый из них указывает, что символ используется как заголовок утверждений; второй - определяет о.д.з. "множество(A)" для утверждения $x \in A$. Справочник "типданных" определяет набор типов операндов отношения "принадлежит", допустимых согласно о.д.з. Он нужен в интерфейсе предварительной обработки задач для ввода посылок и условий, уточняющих тип значений переменных. Справочник "вычисл" позволяет компилятору обрабатывать антецеденты вида "$x \in \{m, \ldots, n\}$", выделенные указателем "программа". Он связывает символ "принадлежит" с оператором "симвномера", перечисляющим символьные номера из заданного промежутка целых чисел.

Свертка дизъюнкции условий принадлежности

Дизъюнкция условий принадлежности неизвестной x известным множествам преобразуется в условие принадлежности объединению этих множеств. Теорема приема имеет вид $\forall_{abc}(a \in (b \cup c) \leftrightarrow (a \in b \vee a \in c))$. Указатель "дизъюнктоперанд" блокирует попытки идентификации с преобразованием конъюнкции в отрицание дизъюнкции отрицаний. Прием не применяется, если задача имеет цель "пример" - здесь выгоднее рассматривать дизъюнктивные члены по отдельности.

Выдача ответа задачи на описание

Если в задаче на описание остается единственное условие $a \in b$, содержащее неизвестную a, причем эта неизвестная не входит в выражение b, то предпринимается исключение a. Прием, выполняющий это действие, имеет теорему $a \in b$ и заголовок "ответзадачи". Его указатель "смответ(1)" означает, что точка привязки берется в первом (и единственном) конъюнктивном члене теоремы; указатель "исключнеизв(x1)" определяет исключаемую неизвестную a. Прием обращается к процедуре "исключнеизв", создающей вспомогательную задачу Z'. Она получается отбрасыванием условия $a \in b$ и удалением неизвестной a. Утверждение $a \in b$ регистрируется в комментарии (контекст $\ldots$) задачи Z' и извлекается из него при завершающем редактировании ответа.

Фильтр "не(цель(коорд))" блокирует применение приема в таких задачах, где условие принадлежности неизвестной точки некоторому геометрическому множеству задано с самого начала, а нужно выразить эту точку через систему числовых параметров.

Ответ выдается также в случае единственного условия $\neg(a \in b)$, где a - неизвестная; b - известное выражение. Теорема приема здесь имеет вид "явное(a набор(не(принадлежит(a b)))пустоеслово пустоеслово)". Допускается наличие прочих условий, не содержащих неизвестных.

Параметрическое описание условия принадлежности

Чтобы получить описание семейства множеств, содержащих заданный элемент, используется прием, основанный на теореме $\forall_{bc}(c-\text{set} \rightarrow (b \in c \leftrightarrow \exists_a(c = a \cup \{b\} \;\&\; a-\text{set}))$. Он срабатывает на достаточно высоком уровне (равном 5) в задаче на описание, имеющей цель "пример" либо "попыткапараметризации". Переменная c - неизвестная задачи; выражение b этой неизвестной не содержит.

Подбор примера

Чтобы выбирать пример точки, принадлежащей числовому промежутку, введен прием, основанный на теореме $\forall_{ax}(0 < b - a \;\&\; x = (a+b)/2 \rightarrow x \in [a,b])$. Промежуток здесь берется с произвольными условиями на свои концы. Контекст срабатывания определяется фильтрами "уровень(1)", "условие", "тип(описать)", "цель(пример)", "неизвестная(x)", "не(контекст(новоеусловие(x5)входит(x x5)))", "известно(a)", "известно(b)". Во вспомогательной задаче условие принадлежности промежутку заменяется на второй антецедент; предварительно устанавливается истинность первого.

Хотя приемы, выбирающие пример элемента для множеств других типов, пока не понадобились, они, без всякого сомнения, будут впоследствии нужны. Сразу ясно, что таких приемов будет много. Эта ситуация хорошо иллюстрирует текущее состояние обучения решателя - в большинстве случаев его база приемов содержит лишь единичные примеры, обозначающие для дальнейшей обработки то или иное явление. Видимо, не всегда будет нужно создавать вручную и хранить все мыслимые варианты приемов - при развитом генераторе приемов их лучше создавать автоматически по мере надобности на основе базы теорем, а затем выбрасывать.

Исключение квантора

Утверждение вида $\exists_b(b-\text{set} \;\&\; a \in b)$ заменяется на константу "истина". Утверждение $\exists_x(x \in a)$ заменяется на $\neg(a = \emptyset)$.

Переход от условия непринадлежности к условию непересечения

Условие непринадлежности известного элемента a неизвестному множеству b удобно переформулировать в терминах непересечения последнего с одноэлементным множеством $\{a\}$. Это позволяет далее использовать общий аппарат решения уравнений в неизвестных множествах.

Усмотрение принадлежности либо непринадлежности с помощью проверочных операторов

Чтобы обеспечивать быстрое усмотрение условия принадлежности, создан проверочный оператор "усмпринадлежит" (см. ниже). Обращения к нему выполняются при сканировании задачи, если в условии обнаружено утверждение вида $a \in b$. Предусмотрено также обращение для посылок, однако оно относится к крайне редкой ситуации. Теорема приема имеет вид $\forall_{ab}(a \in b \to a \in b)$. Антецедент обрабатывается проверочным оператором "усмпринадлежит". Указатель "новый" означает, что попытки усмотреть принадлежность будут происходить не только при первом возникновении текущего условия, но и при каждом изменении задачи. Это важно, так как вновь выведенные посылки или условия могут сделать очевидной ранее не усматривавшуюся принадлежность. В случае условия задачи на описание, содержащего неизвестные, уровень обращения к приему несколько выше - обычно здесь работают другие средства.

Аналогичный прием введен для усмотрения непринадлежности. Он использует проверочный оператор "усмнепринадлежит".

Усмотрение различия элементов из условий принадлежности и непринадлежности

Если в задаче есть равенство $b = c$, контекст которого содержит утверждение $b \in a$, то предпринимается обращение к проверочному оператору для усмотрения утверждения $\neg(c \in a)$. В случае успеха равенство заменяется на константу "ложь". Два аналогичных приема введены для усмотрения ложности равенств $b-c = 0$ и $c-b = 0$.

Свертка двух неизвестных условий непринадлежности в одно

Два утверждения о непринадлежности одной и той же неизвестной x известным множествам a, b объединяются в одно утверждение о непринадлежности объединению этих множеств. Введены две версии приема. Одна из них имеет заголовок "второйтерм" и рассчитана на появление условий непринадлежности под общей конъюнкцией. Другая имеет заголовок "заменаусловия(второйтерм)" и рассчитана на случай двух независимых условий задачи.

Проверочный оператор "усмпринадлежит"

Оператор предназначен для проверки утверждений вида $a \in b$. Реализация его приемов на ГЕНОЛОГе крайне проста, поэтому ограничимся лишь указанием теорем и краткими пояснениями.

1. Использование включения множества. $\forall_{abc}(a \subseteq b \; \& \; c \in a \to c \in b)$. Имеются две версии приема. В первой из них включение принадлежит контексту, а принадлежность проверяется проверочным оператором. Во второй - наоборот, в контексте усматривается принадлежность, а включение проверяется. Кроме того, введены аналогичные приемы для включения в прямое произведение. Теоремы их суть $\forall_{abcde}(a \subseteq b \times c \; \& \; (d, e) \in a \to e \in c)$; $\forall_{abcde}(a \subseteq b \times c \; \& \; (d, e) \in e \to d \in b)$. Все антецеденты должны быть представлены в контексте явным образом.

2. Принадлежность объединению. $\forall_{abc}(a \in b \to a \in b \cup c)$. Антецедент обрабатывается проверочным оператором.

3. Принадлежность пересечению. $\forall_{abc}(a \in b \; \& \; a \in c \to a \in b \cap c)$. Указатель "дистрибразвертка(фикс(0 2))" определяет обработку сразу всех членов пересечения. Указатель "спуск" означает, что в случае неприменения данного приема сразу выдается отказ.

4. Принадлежность разности. $\forall_{abc}(\neg(a \in c)\ \&\ a \in b \to a \in b \setminus c)$.

5. Принадлежность конечному списку. $\forall_{ab}(a \in \{a; b\})$. Запись $\{a; b\}$ означает "перечень(префикс(a b))". Указатель "список(фикс(0 2 1))" определяет идентификацию переменной a с произвольным (не обязательно первым) элементом конечного списка.

6. Принадлежность представителя непустого множества этому множеству. В логическом языке предусмотрен символ "элемент(a)", выделяющий какой-то элемент непустого множества a. Об этом элементе, кроме принадлежности его множеству a, ничего не известно. Для усмотрения данной принадлежности служит прием, основанный на теореме $\forall_a(a - \mathrm{set}\ \&\ \neg(a = \emptyset) \to \text{элемент}(a) \in a)$.

7. Принадлежность образу множества. $\forall_{afx}(x \in a \to f(x) \in \text{образ}(f, a))$.

8. Принадлежность конечному отрезку целых чисел. Для усмотрения такой принадлежности созданы несколько приемов. Прежде всего, это прием, основанный на теореме $\forall_{abn}(n - \text{целое}\ \&\ 0 \le (n - a)\ \&\ 0 \le (b - n) \to n \in \{a, \ldots, b\})$. Все три антецедента обрабатываются проверочными операторами. Если в посылках уже имеется утверждение о принадлежности элемента a конечному отрезку $\{b, \ldots, c\}$, то для проверки принадлежности его конечному отрезку $\{d, \ldots, e\}$ служит прием, основанный на теореме $\forall_{abcde}(a \in \{b, \ldots, c\}\ \&\ 0 \le (b - d)\ \&\ 0 \le (e - c) \to a \in \{d, \ldots, e\})$. Второй и третий антецеденты обрабатываются проверочными операторами. Следующие два приема относятся к ситуации, когда утверждение принадлежности в посылках относится к элементу, отличающемуся от данного. Теоремы их таковы: $\forall_{amnkpq}(a \in \{p, \ldots, q\}\ \&\ p = m - k\ \&\ q = n - k\ \&\ k - \text{целое} \to (a + k) \in \{m, \ldots, n\})$; $\forall_{imnrskp}(i \in \{m, \ldots, n\}\ \&\ 0 \le (km + p - r)\ \&\ 0 \le (s - kn - p)\ \&\ k - \text{целое}\ \&\ p - \text{целое} \to (ki + p) \in \{r, \ldots, s\})$. Уровень срабатывания первого приема равен 2, второго - 3. Часто условие принадлежности целого числа конечному отрезку преобразуется в два неравенства. В этой ситуации используется следующая модификация предыдущего приема: $\forall_{imnrskp}(m \le i\ \&\ i \le n\ \&\ 0 \le (km + p - r)\ \&\ 0 \le (s - kn - p)\ \&\ k - \text{целое}\ \&\ p - \text{целое} \to (ki + p) \in \{r, \ldots, s\})$. Здесь первые два антецедента явно идентифицируются с посылками, остальные - проверяются. Наконец, имеется прием, основанный на усмотрении в посылках утверждений $a \in \{m, \ldots, n\}$ и $a < b$. Он проверяет принадлежность b тому же отрезку $\{m, \ldots, n\}$ и делает вывод о принадлежности a отрезку $\{m, \ldots, n - 1\}$.

9. Принадлежность промежутку. Перечисляются приемы, сводящие условия принадлежности числовому промежутку к соответствующим неравенствам. Случаи принадлежности лучу или всей числовой прямой рассматриваются отдельно. Приведем единственный пример - прием $\forall_{abc}(b - \text{число}\ \&\ c - \text{число}\ \&\ a - \text{число}\ \&\ 0 < (a - b)\ \&\ 0 < (c - a) \to a \in (b, c))$. Все антецеденты обрабатываются проверочными операторами. Теоремы остальных приемов имеют аналогичный вид.

10. Принадлежность классу. Чтобы усматривать принадлежность множеству, определенному с помощью описателя "класс", реализуется обращение к вспомогательной задаче на доказательство. Теорема приема имеет вид $\forall_{fa}(f(a) \to a \in \mathrm{set}_x(f(x)))$. Указатель "легковидеть(1)" определяет проверку антецедента с помощью задачи, решаемой до максимального уровня 4. Указатель "лимит(...)"

ограничивает трудоемкость обращения так, чтобы попытка была сравнительно быстрой.

11. Усмотрение принадлежности области значений отображения. $\forall_{ABfc}$(Отображение(f, A, B) & $c \in A \rightarrow f(c) \in B)$.

12. Усмотрение принадлежности прямому произведению. Предусмотрены два приема, основанные на теоремах: $\forall_{abf}(l(f) = 2$ & $f(1) \in a$ & $f(2) \in b \rightarrow f \in a \times b)$; $\forall_{abcd}(a \in c$ & $b \in d \rightarrow (a, b) \in c \times d)$. Первый антецедент первой теоремы выделен указателем "идентификатор". Выражение $l(f)$ для длины набора f обрабатывается нормализатором "нормдлинанабора", и проверяется, что эта длина равна 2.

13. Принадлежность сомножителю прямого произведения. $\forall_{acde}(a \subseteq (d \times e)$ & $c \in a \rightarrow c(1) \in d)$. Аналогичный прием введен для второго элемента пары c. Первый антецедент явным образом присутствует в контексте.

14. Прочие приемы. В пакете, кроме перечисленных выше, представлены еще около двух десятков приемов для принадлежности специальным множествам, например, геометрическим фигурам и временным промежуткам процессов. Реализация их на ГЕНОЛОГе не содержит каких-либо особенностей.

Проверочный оператор "усмнепринадлежит"

Оператор совершенно аналогичен предыдущему. Например, прием, усматривающий непринадлежность объединению множеств, имеет теорему $\forall_{abc}(\neg(a \in b)$ & $\neg(a \in c) \rightarrow \neg(a \in (b \cup c)))$. Антецеденты обрабатываются проверочными операторами; чтобы сразу обрабатывать все операнды многоместного объединения, введен указатель "дистрибразвертка(фикс(0 1 2))".

Нормализатор уравнений "уравнпринадлежит"

Нормализатор предназначен для ускоренного разрешения относительно неизвестных выделенного условия задачи на описание и аналогичен рассмотренному ранее нормализатору "уравнмножество". Перечислим его приемы. Все они, при определенных предположениях о контексте преобразований, обеспечивают расшифровку ("развертку") условий принадлежности по определениям.

1. Развертка условия принадлежности объединению. $\forall_{abc}(a \in (b \cup c) \leftrightarrow (a \in b \vee a \in c))$. Предполагается, что либо a не содержит неизвестных, либо хотя бы одно из множеств b, c - содержит. Каждое из условий принадлежности операндам обрабатывается тем же самым нормализатором, а их дизъюнкция - нормализатором "нормлог". Если объединение имеет несколько операндов, то данный прием выбирает один из них - множество b, а объединение остальных операндов играет роль множества c. Рекурсивные обращения обеспечивают подключение всех имеющихся ресурсов для разрешения утверждения относительно неизвестных.

2. Развертка условия принадлежности пересечению. $\forall_{abc}(a \in (b \cap c) \leftrightarrow (a \in b$ & $a \in c))$. Фильтры и указатели - аналогично предыдущему приему.

3. Развертка условия принадлежности разности. $\forall_{abc}(a \in (b \setminus c) \leftrightarrow (a \in b \ \& \ \neg(a \in c)))$.

4. Развертка условия принадлежности конечному списку. $\forall_{abc}(a \in \{b; c\} \leftrightarrow (a = b \ \vee \ a \in \{; c\}))$.

5. Развертка условия принадлежности прямому произведению. $\forall_{abf}(f \in a \times b \leftrightarrow (l(f) = 2 \ \& \ f(1) \in a \ \& \ f(2) \in b \ \& \ f$ – слово$))$. Единственное ограничение на контекст - отсутствие неизвестных в выражении f.

6. Развертка условия принадлежности прообразу неизвестного множества. $\forall_{afx}(x \in$ прообраз$(f, a) \leftrightarrow (x \in \mathrm{Dom}(f) \ \& \ f(x) \in a))$. Выражение a содержит неизвестные; выражения f, x - не содержат.

7. Развертка условия принадлежности прообразу неизвестного элемента. $\forall_{afx}(x \in$ слой$(f, a) \leftrightarrow a = f(x))$. Ограничения на контекст - те же, что в предыдущем приеме.

Нормализатор "нормпринадлежит"

Расшифровка по определениям условия принадлежности бывает нужна не только при разрешении утверждения относительно неизвестных, но и во многих других случаях. Чтобы ускорить ее, используется нормализатор "нормпринадлежит". Приемы этого нормализатора аналогичны приемам нормализатора "уравнпринадлежит" и проверочного оператора "усмпринадлежит". В них легко разобраться непосредственно по описаниям ГЕНОЛОГа. Заметим, что такого рода нормализатор общей стандартизации утверждений - явление достаточно редкое.

Синтезатор "выборточки"

Чтобы ускорить выбор произвольного элемента множества, создан синтезатор "выборточки". Однако, он заполнен пока лишь символически - имеет три приема, ориентированные на ту задачу, ради которой синтезатор и был создан.

4.4 Приемы символа "содержится"

Общая характеризация символа обеспечивается справочниками "предикатныйсимвол", "одз", "типданных", "бинарноеотношение", "транзитивно", "монотонно", "минимум", "максимум". Первые три справочника уже рассматривались выше. Справочник "бинарноеотношение" позволяет находить по типу объектов всевозможные бинарные отношения, рассматриваемые между объектами данного типа (в нашем случае - между множествами). Справочник "транзитивно" определяет транзитивность отношения включения; справочник "монотонно" - указывает, что истинность включения сохраняется при монотонном убывании первого операнда или монотонном возрастании второго. Справочники "минимум" и "максимум" указывают операции "пересечение" и "объединение", дающие точную нижнюю и верхнюю грани пары элементов для отношения порядка "содержится".

Кванторная расшифровка условия включения множеств

Если условие задачи на доказательство имеет вид включения либо отрицания включения, то предпринимается его кванторная расшифровка. Для множеств, заданных с помощью описателя "класс", уровень срабатывания приема берется сравнительно маленьким - равным 2. Иначе он равен 6 для условия включения и 7 для отрицания включения. Теоремы обоих приемов одинаковы - $\forall_{ab}(a \subseteq b \leftrightarrow \forall_c(c \in a \rightarrow c \in b))$. Заголовок приема - "второйтерм". После применения кванторной расшифровки вводится комментарий "кванторнаясвертка", блокирующий обратное преобразование. Хотя условие задачи на доказательство, имеющее своим заголовком квантор общности, удерживается недолго - оно расформируется на составные части описанным ранее общелогическим приемом, однако это преобразование выполняется лишь на втором уровне, а прием кванторной свертки может сработать уже на уровне 0.

Кванторная свертка в условие включения множеств

За исключением указанного выше случая условий задачи на доказательство, в остальных ситуациях применяется преобразование кванторной свертки в условие включения. Теорема приема та же, что и при кванторной расшифровке. Однако, теперь заголовок приема "первыйтерм" указывает преобразование справа налево. Уровни срабатывания - 0 либо 2. При этом уровень 2 относится лишь к процессам вывода теорем, на что указывает цель "редуцирование". Указатель "кванторнаясвертка" определяет попытки идентификации с преобразованием квантора существования в отрицание квантора общности и с группировкой всех подкванторных утверждений, зависящих от заданной переменной c.

Подбор примера

Чтобы подбирать пример надмножества заданного множества, создан прием с теоремой $\forall_{ax}(x = a \rightarrow a \subseteq x)$. Здесь x - неизвестная; a известно. Предполагается, что нет других содержащих x условий, кроме, быть может, условия "множество(x)". Напомним, что для подбора примера подмножества имеется прием, выбирающий пустое множество.

Включение множества в себя

Условие включения множества в себя заменяется на константу "истина".

Параметрическое описание для включения множеств

Для параметрического описания семейства надмножеств заданного множества имеется прием, основанный на теореме $\forall_{ab}(a \subseteq b \leftrightarrow \exists_c(c - \text{set} \ \& \ b = a \cup c))$. Здесь b - неизвестная, не входящая в a. Прием применяется, если в задаче нужно найти пример либо параметрическое описание. Уровень срабатывания равен 5. Для параметрического описания подмножеств создан аналогичный прием с теоремой $\forall_{ab}(b \subseteq a \leftrightarrow \exists_c(c - \text{set} \ \& \ b = a \cap c))$. Здесь снова b - неизвестная, не входящая в a. Однако, для множеств специальных типов могут использоваться другие, более удобные параметрические описания их подмножеств. Поэтому данный прием ограничивается случаем, когда a - переменная. Напомним, что приемы с заголовком "параметризация" не выполняют замены условия задачи, а сводят эту задачу к другой, в которой произведена замена условия. При неудаче происходит продолжение решения неизмененной исходной задачи.

Параметрическое описание для невключения множеств

Для невключения множеств используется неявное параметрическое описание, основанное на теореме $\forall_{ab}(\neg(a \subseteq b) \leftrightarrow \exists_c(c \in a \,\&\, \neg(c \in b)))$. Уровень срабатывания равен 6; преобразуемое условие должно содержать неизвестные.

Замена конъюнкции двух включений на равенство множеств

Прием основан на теореме $\forall_{ab}(a = b \leftrightarrow (a \subseteq b \,\&\, b \subseteq a))$. Заголовок приема - "первыйтерм". Он применяется на уровне 2 в любом контексте. Однако, прием не относится к случаю двух различных посылок или условий, представляющих собой встречные включения. Оба включения должны быть конъюнктивными членами одной конъюнкции, либо антецедентами одной импликации, либо отрицания их должны быть дизъюнктивными членами одной дизъюнкции.

Группировка двух посылок - встречных включений

Прием основан на той же теореме, что и предыдущий. Однако заголовок приема - "заменатермов(первыйтерм)" - означает, что встречные включения идентифицируются с двумя различными посылками задачи.

Группировка двух условий - встречных включений

Прием аналогичен предыдущему; его заголовок - "заменаусловия(первыйтерм)". Предусмотрены несколько уровней срабатывания - 0, 2 и 4. Это позволяет реагировать на появление одного из включений с запаздыванием по отношению к другому.

Замена включения множеств на равенство, если из контекста усматривается встречное включение

Теорема $\forall_{ab}(a \subseteq b \rightarrow (a = b \leftrightarrow b \subseteq a))$ определяет прием, предпринимающий проверку встречного включения проверочным оператором. Он имеет заголовок "первыйтерм". Если включение находится в условии задачи на доказательство, уровень срабатывания равен 3. Иначе он равен 0. Если одно из выражений a, b - переменная, входящая в другое выражение, применение приема блокируется.

Исключение квантора существования

Утверждения существования надмножества либо подмножества заменяются на константу "истина". Утверждение существования множества x, содержащего a и содержащегося в b, заменяется на включение множества a в множество b.

Выдача ответа задачи на описание

Предусмотрены два шаблона для усмотрения ответа, связанного с включением множеств. Первый из них состоит из утверждений $x - \mathrm{set}$, $a \in x$, $x \subseteq b$, быть может, пополненных произвольным числом утверждений вида $\neg(x \subseteq c)$, $\neg(c \subseteq x)$. Ответом относительно неизвестной x считается произвольное подмножество такого шаблона.

При выдаче ответа вводится дополнительное утверждение $a \in b$, накладывающее на известные объекты a, b условие непустоты множества значений неизвестной. Как и обычно, предусмотрены несколько приемов усмотрения ответа для заданного шаблона, соответствующих инициализации по различным элементам шаблона.

Второй шаблон ответа состоит из утверждений $x - \mathrm{set}$, $a \subseteq x$, $x \subseteq b$, быть может, пополненных утверждениями вида $\neg(x \subseteq c)$, $\neg(d \subseteq x)$.

Вывод следствий в посылках

Если в посылках явно указаны некоторые включения, то каждое утверждение о принадлежности множеству может инициировать, согласно данным посылкам, вывод следствий о принадлежности его надмножествам. Однако, такой вывод применяется лишь в особых случаях. Имеются два приема с одной и той же теоремой: $\forall_{abc}(a \in b \ \& \ b \subseteq c \to a \in c)$. Первый применяется, если решается задача на доказательство, условие которой имеет вид $P(a)$. Уровень срабатывания его равен 4. Второй применяется, если решается задача на исследование, имеющая цель "известно". Это - ситуация анализа объединенного списка посылок и условий задачи на вычисление. Уровень срабатывания здесь равен 2.

Проверка включения или невключения с помощью проверочных операторов

Для проверки включения и невключения созданы операторы "усмсодержится" и "усмнесодержится". Для обращения к первому из них имеются два приема, основанные на теореме $\forall_{ab}(a \subseteq b \to a \subseteq b)$. Антецедент обрабатывается оператором "усмсодержится". При успехе включение заменяется на константу "истина". Если включение является условием задачи на доказательство, то применяется первый прием. Уровни срабатывания его равны 1 и 3. Другой прием, имеющий уровни срабатывания 2 и 4, применяется, если задача не имеет типа "доказать". Если включение входит в условие задачи на описание, содержащей неизвестные, то уровень срабатывания здесь равен 4, иначе он равен 2.

Для обращения к оператору "усмнесодержится" имеется единственный прием аналогичного вида.

Проверочный оператор "усмсодержится"

Перечислим основные приемы оператора. Они столь же просто реализуются на ГЕНОЛОГе, как и приемы оператора "усмпринадлежит". Поэтому, как правило, ограничимся указанием теорем.

1. Включение множества в себя. $\forall_a(a \subseteq a)$.
2. Включение пустого множества. $\forall_a(\emptyset \subseteq a)$.
3. Транзитивность. $\forall_{abc}(a \subseteq b \ \& \ b \subseteq c \to a \subseteq c)$. Имеются две версии приема. В одной из них первый антецедент входит в посылки, а второй проверяется, в другой - наборот.
4. Включение в объединение. $\forall_{abc}(a \subseteq b \to a \subseteq (b \cup c))$;

   $\forall_{bcef}(c \subseteq b \cup e \ \& \ c \subseteq b \cup f \to c \subseteq (e \cap f) \cup b)$;

   $\forall_{bcef}(c \subseteq b \cup e \ \& \ \text{непересек}(f \setminus b, c) \to c \subseteq (e \setminus f) \cup b)$.

5. Включение объединения. $\forall_{abc}(a \subseteq c \;\&\; b \subseteq c \to a \cup b \subseteq c)$. Указатель "дистрибразвертка(фикс(0 1))" определяет одновременную обработку всех членов объединения.

6. Включение пересечения. $\forall_{abc}(b \subseteq a \to b \cap c \subseteq a)$;

   $\forall_{bcef}(b \cap e \subseteq c \;\&\; b \cap f \subseteq c \to (e \cup f) \cap b \subseteq c)$.

7. Включение в пересечение. $\forall_{abc}(c \subseteq a \;\&\; c \subseteq b \to c \subseteq (a \cap b))$.

8. Включение разности. $\forall_{abc}(a \subseteq (b \cup c) \to (a \setminus b) \subseteq c)$.

9. Включение в разность. $\forall_{abc}(c \subseteq a \;\&\; \text{непересек}(b, c) \to c \subseteq (a \setminus b))$.

10. Включение конечного списка. $\forall_{abc}(\{; b\} \subseteq c \;\&\; a \in c \to \{a; b\} \subseteq c)$. Указатель "спуск" блокирует альтернативные попытки усмотрения включения. Рекурсия приводит к последовательной проверке принадлежности элементов списка множеству c.

11. Включение симметрической разности. $\forall_{abcd}(b \bigtriangleup c \subseteq (a \bigtriangleup b) \cup (a \bigtriangleup c) \cup d)$.

12. Включение прямых произведений. $\forall_{abcd}(b \subseteq d \;\&\; a \subseteq c \to (a \times b \subseteq c \times d))$.

13. Включение образа. $\forall_{abc}(a \subseteq b \to \text{образ}(c, a) \subseteq \text{образ}(c, b))$;

    $\forall_{af}(\text{образ}(f, \text{прообраз}(f, a)) \subseteq a)$;

    $\forall_{cf}(\text{образ}(f, c) \subseteq \mathrm{Val}(f))$;

    $\forall_{abcf}(\text{образ}(f, a) \subseteq c \;\&\; \text{образ}(f, b) \subseteq c \to \text{образ}(f, a \cup b) \subseteq c)$;

    $\forall_{abcf}(\text{образ}(f, a) \subseteq c \to \text{образ}(f, a \cap b) \subseteq c)$;

    $\forall_{abcf}(\text{образ}(f, a) \subseteq c \to \text{образ}(f, a \setminus b) \subseteq c)$;

    $\forall_{abf}(b \subseteq \text{прообраз}(f, a) \to \text{образ}(f, b) \subseteq a)$.

14. Включение прообразов. $\forall_{abc}(a \subseteq b \to \text{прообраз}(c, a) \subseteq \text{прообраз}(c, b))$.

15. Включение прообраза в область определения. $\forall_{ab}(\text{прообраз}(b, a) \subseteq \mathrm{Dom}(b))$.

16. Включение в прообраз. $\forall_{abf}(b \subseteq \mathrm{Dom}(f) \;\&\; \text{образ}(f, b) \subseteq a \to b \subseteq \text{прообраз}(f, a))$.

17. Включение двух классов. $\forall_{fg}(\mathrm{set}_x(f(x) \;\&\; g(x)) \subseteq \mathrm{set}_y(f(y)))$.

18. Включение класса. $\forall abf(a \subseteq b \to \mathrm{set}_x(x \in a \;\&\; f(x)) \subseteq b)$;

    $\forall_{abcf}(c = \mathrm{set}_x(x \in a \;\&\; f(x)) \;\&\; a \subseteq b \to c \subseteq b)$;

    $\forall_{afb}(a = \mathrm{set}_x(f(x)) \;\&\; \mathrm{set}_x(f(x)) \subseteq b \to a \subseteq b)$. В последних двух случаях первый антецедент находится в посылках, а второй проверяется.

    $\forall_{aP}(x \in a \to \mathrm{set}_x(P(x)) \subseteq a)$. Здесь утверждение $P(x)$ не содержит кванторов. Первый антецедент обрабатывается проверочным оператором. Указатель "занесениепосылки(1 значение(x40 x23))" присоединяет к списку посылок этого оператора конъюнктивные члены утверждения $P(x)$.

    $\forall_{faA}(f(y) \in a \to \mathrm{set}_x(\exists_y(x = f(y) \;\&\; A(y))) \subseteq a)$. Прием аналогичен предыдущему.

19. Включение в множество чисел. Для проверки включения в множества вещественных либо целых чисел имеется несколько простых приемов (например, $N \subseteq R$). Напомним, что множество вещественных чисел во внутреннем представлении записывается как "промежуток(минусбеск плюсбеск 0 0)", а прорисовывается формульным редактором как R.

20. Включение конечных отрезков целых чисел.

    $\forall_{mnpq}(m - \text{целое} \ \& \ n - \text{целое} \ \& \ p - \text{целое} \ \& \ q - \text{целое} \ \& \ 0 \leq p - m \ \& \ 0 \leq n - q \rightarrow \{p, \dots, q\} \subseteq \{m, \dots, n\})$.

21. Включение пересечения семейства.

    $\forall_{ABmn}(0 \leq n - m \ \& \ B(m) \subseteq A \rightarrow \bigcap_{i=m}^{n} B(i) \subseteq A)$;

    $\forall_{AjP}(P(j) \rightarrow \bigcap_{i,P(i)} A(i) \subseteq A(j))$.

    Антецедент обрабатывается вспомогательной задачей на доказательство, решаемой с ограничениями на уровень и трудоемкость.

22. Использование кванторной посылки. Следующие два приема используют кванторную посылку, устанавливающую включение для некоторых семейств множеств.

    $\forall_{mnkAB}(\forall_i(i \in \{m, \dots, n\} \rightarrow B(i) \subseteq A) \ \& \ k \in \{m, \dots, n\} \rightarrow B(k) \subseteq A)$.

    Первый антецедент имеется в посылках. Выражение k идентифицируется путем сопоставления известного шаблона $B(i)$ и левой части проверяемого включения. Затем проверяется второй антецедент.

    $\forall_{mnkABab}(a = A(k) \ \& \ b = B(k) \ \& \ \forall_i(i \in \{m, \dots, n\} \rightarrow B(i) \subseteq A(i)) \ \& \ k \in \{m, \dots, n\} \rightarrow b \subseteq a)$.

    Третий антецедент берется в посылках. Первые два помечены указателями "идентификатор". Они обеспечивают идентификацию k путем сопоставления $A(i), B(i)$ с частями проверяемого включения. Четвертый антецедент обрабатывается проверочным оператором. Попытка применения приема выполняется лишь при условии, что проверяемое включение содержит символ "значение", так как обычно кванторные посылки указанного вида относятся к каким-либо рассматриваемым в задаче функциям.

23. Числовые промежутки.

    $\forall_{abcdef}(0 \leq c - a \ \& \ 0 \leq b - d \rightarrow [c, d] \subseteq [a, b])$.

    Во внутреннем представлении указатели типа концов обоих промежутков одинаковы. В фильтрах указано, что одноименные концы промежутков различаются. Для случаев, когда они совпадают, созданы несколкьо дополнительных приемов.

Проверочный оператор "усмнесодержится"

Оператор аналогичен предыдущему и не имеет каких-либо новых особенностей. Рекомендуется просмотреть его приемы самостоятельно.

Нормализатор "уравнсодержится"

Этот нормализатор, как и рассмотренные выше нормализаторы "уравнмножество", "уравнпринадлежит", используется при явном разрешении единственного уравнения относительно неизвестных множеств. Так как он выполняет достаточно большой объем полезной работы, остановимся на его приемах несколько подробнее.

1. Развертка включения объединения. $\forall_{abc}(a \cup b \subseteq c \leftrightarrow (a \subseteq c \ \& \ b \subseteq c))$. Либо объединение содержит неизвестную, либо выражение c не содержит неизвестных. В первом случае делается шаг к явному разрешению относительно неизвестных (глубина их вхождения в условия уменьшается), во втором случае предпринимается общая декомпозиция "известного" утверждения для получения более явного контекста. Оба включения в правой части обрабатываются тем же самым нормализатором "уравнсодержится", а вся конъюнкция - нормализатором "нормлог". Таким образом, результатом применения приема будет окончательный результат попытки явного разрешения относительно неизвестных. Напомним, что нормализатор "уравнсодержится" (и другие нормализаторы уравнений) корневой. Это означает, что как только преобразуемое утверждение не будет иметь вида включения, преобразования завершатся и будет выдан ответ.

2. Развертка включения в пересечение. $\forall abc(c \subseteq a \cap b \leftrightarrow (c \subseteq a \ \& \ c \subseteq b))$. Прием аналогичен предыдущему.

3. Развертка включения в разность. $\forall_{abc}(c \subseteq a \setminus b \leftrightarrow c \subseteq a \ \& \ \text{непересек}(b, c))$. Аналогично предыдущему.

4. Развертка включения конечного списка. $\forall_{abc}(\{a; b\} \subseteq c \leftrightarrow (\{; b\} \subseteq c \ \& \ a \in c))$. Здесь несколько изменяется фильтр. Как и выше, прием может быть применен, если конечный список содержится неизвестные либо выражение c не содержит неизвестных. Однако, добавляются еще три возможности - заголовки "объединение", "перечень", "прямоепроизведение" выражения c. В этих случаях декомпозиция утверждения относительно неизвестных может быть продолжена для утверждения $a \in c$.

5. Развертка включения прямых произведений. $\forall_{abcd}(a \times b \subseteq c \times d \leftrightarrow (a \subseteq c \ \& \ b \subseteq d \ \vee \ a = \emptyset \ \vee \ b = \emptyset))$. Прием применяется всегда, так как выполняет общую стандартизацию.

6. Развертка включения пересечения с конечным списком. $\forall_{abcd}(a \cap \{b; c\} \subseteq d \leftrightarrow (\neg(b \in a) \ \vee \ b \in d) \ \& \ a \cap \{; c\} \subseteq d)$. Либо конечный список содержит неизвестные, либо каждое из выражений a, d, содержащее неизвестные, имеет специальный заголовок. Для a такими заголовками являются символы "объединение", "разность", "прямоепроизведение"; для d - символы "объединение", "перечень", "прямоепроизведение". Наличие этих заголовков позволяет продолжить применение преобразований, направленных на явное разрешение относительно неизвестных.

7. Переход от включения разности к включению в объединение.

   $\forall_{abc}(a \subseteq b \cup c \leftrightarrow a \setminus b \subseteq c)$.

   Прием применяется, если выражение c не содержит неизвестных.

8. Приемы для разрешения относительно неизвестной. Одна из переменных считается неизвестной; все остальные - известными.

    (a) Включение в объединение. $\forall_{abc}(a \subseteq a \cup c \leftrightarrow b \setminus c \subseteq a)$. Предпринимается явное разрешение относительно содержащего неизвестные выражения a при известных b, c. Указатель "перечень(x1 не(известно(x1)))" относит к a все члены с неизвестной. Так как для c нет указателя "единица", должен остаться хотя бы один член без неизвестных.

    (b) Включение пересечения. $\forall_{abc}($непересек$(c, a \setminus b) \leftrightarrow a \cap c \subseteq b)$. c содержит неизвестные; a, b - известны. Эквивалентность применяется справа налево.

    (c) Включение разности. $\forall_{abc}(b \setminus a \subseteq c \leftrightarrow b \setminus c \subseteq a)$. Аналогично предыдущему.

    (d) Включение в прообраз. $\forall_{abc}(b \subseteq \mathrm{Dom}(c) \rightarrow$ образ$(c, b) \subseteq a \leftrightarrow b \subseteq$ прообраз$(c, a))$. Эквивалентность применяется справа налево. Выражение a содержит неизвестные; b, c известны.

    (e) Включение образа. $\forall_{abc}(b \subseteq \mathrm{Dom}(c) \rightarrow$ образ$(c, b) \subseteq a \leftrightarrow b \subseteq$ прообраз$(c, a))$. Эквивалентность такая же, как в предыдущем приеме, но применяется слева направо. Выражение b содержит неизвестные; a, c известны.

9. Приемы группировки неизвестных в одной части уравнения. Два выражения с неизвестными, ранее располагавшиеся в разных частях включения, группируются в одной части. После этого к ним применяется подходящий нормализатор разрешения относительно неизвестных. Он сопровождается комментарием "группировка", блокирующим повторные попытки применения приемов данного подраздела.

    (a) Переход от включения к равенству разности пустому множеству. $\forall_{ab}(a \setminus b = \emptyset \leftrightarrow a \subseteq b)$. Каждое из выражений a, b содержит неизвестные. Эквивалентность применяется справа налево. Сначала разность множеств a, b обрабатывается нормализаторами "нормразность", "уравнразность". Первый обеспечивает общую стандартизацию, второй - упрощение выражения относительно неизвестных. Такое упрощение может заключаться, например, в приведении подобных неизвестных членов. После нормализации разности ко всему заменяющему терму применяется нормализатор "уравнмножество". Он доводит до конца начатую цепочку преобразований по разрешению относительно неизвестных. Указатель "длина(Контрольглубины)" контролирует уменьшение глубины неизвестных.

    (b) Переход от включения в объединение к включению разности. $\forall_{abc}(a \setminus b \subseteq c \leftrightarrow a \subseteq b \cup c)$. Эквивалентность применяется справа налево. Выражения a, b содержат неизвестные; на c ограничений не накладывается. В остальном прием устроен аналогичено предыдущему, но вместо нормализатора "уравнмножество" левая часть обрабатывается нормализатором "уравнсодержится".

    (c) Переход от включения разности к включению в объединение. Теорема та же, что у предыдущего приема. Однако, преобразование выполняется слева направо. Выражения b, c содержат неизвестные.

    (d) Переход от включения пересечения к условию непересечения. $\forall_{abc}($непересек$(c, a \setminus b) \leftrightarrow a \cap c \subseteq b)$. Эквивалентность применяется справа налево; выражения a, b содержат неизвестные.

   (e) Переход от включения в прообраз к включению образа. $\forall_{abc}(b \subseteq \mathrm{Dom}(c) \rightarrow (\text{образ}(c,b) \subseteq a \leftrightarrow b \subseteq \text{прообраз}(c,a)))$. Эквивалентность применяется справа налево; выражения b, c содержат неизвестные.

   (f) Переход от включения образа к включению в прообраз. Теорема та же, что у предыдущего приема, но применяется слева направо. Выражения a, c содержат неизвестные.

10. Попытка представить левую часть включения в виде объединения множеств.

    Если левая часть включения имеет вид объединения, то оно распадается на несколько более простых включений Поэтому можно существенно продвинуться в разрешении включения относительно неизвестных, если преобразовать его левую часть к виду объединения пересечений. Такое преобразование выполняется нормализатором "видобъединение". Прием, обращающийся к нему, имеет теорему вида $\forall_{cda}(a = d \rightarrow d \subseteq c \leftrightarrow a \subseteq c)$. Антецедент выделен указателем "идентификатор(1)". Он присваивает переменной a результат обработки выражения d нормализатором "видобъединение". Заменяющее утверждение $a \subseteq c$ обрабатывается нормализатором "уравнсодержится". Прием применяется, если выражение d содержит неизвестные, отлично от переменной и не имеет заголовка "объединение". Проверяется, что выражение a либо имеет заголовок "объединение", либо короче выражения d.

11. Попытка представить правую часть включения в виде пересечения либо разности. Прием аналогичен предыдущему. Для декомпозиции включения он пытается представить правую часть в виде пересечения либо разности с помощью нормализатора "видразность". Теорема имеет вид $\forall_{cda}(a = d \rightarrow c \subseteq d \leftrightarrow c \subseteq a)$.

12. Попытка представить одну из частей включения в виде прямого произведения, если другая часть уже имеет вид прямого произведения. Для этой цели используются два приема. Теорема первого из них $\forall_{cdea}(a = e \rightarrow e \subseteq c \times d \leftrightarrow a \subseteq c \times d)$ относится к случаю, когда прямое произведение имеется справа от включения. Если оно расположено слева от включения, применяется второй прием. В обоих случаях используется нормализатор "видпрямоепроизведение". Включение должно содержать неизвестные; выражение e не является переменной и не имеет заголовка "прямоепроизведение".

4.5 Приемы символа "объединение"

Начнем с рассмотрения приемов справочников для символа "объединение". Справочники "арность", "одз", "типданных", "тип" указывают, что операция двуместная, определена на множествах и значениями ее служат множества. Операция выделена справочниками "коммутативно" и "ассоциативно". Справочник "единица" указывает единицу операции - пустое множество. Справочник "область" констатирует, что операция принимает любое значение из своей области определения. Он основан на теореме $\forall_a(a - \mathrm{set} \rightarrow a \cup a = a)$, указывающей, какие именно знчения операндов дают требуемое значение операции. Данный справочник бывает полезен при заменах подвыражений вида $x \cup y$ на новые переменные. Справочник "дистрибутивно" указывает на дистрибутивность объединения относительно пересечения. Наконец, используемый компилятором ГЕНОЛОГа справочник "вычисл" указывает на возможность применения оператора "объединениесписков" для нахождения объединения двух конечных списков объектов. Эти объекты должны быть некоторыми вычислительными структурами данных ГЕНОЛОГа.

На теореме "коммутативно(объединение)" основаны прием, устраняющий вложенные объединения и прием, лексикографически упорядочивающий операнды.

Общая стандартизация выражений

Несколько простых приемов выполняют общую стандартизацию объединений:

1. Объединение с пустым множеством. $\forall_a(a \cup \emptyset = a)$;

2. Объединение одинаковых множеств. $\forall_a(a \cup a = a)$;

3. Безусловные поглощения. $\forall_{ab}(a \cup (a \cap b) = a)$; $\forall_{ab}(a \cup (a \setminus b) = a)$;

4. Условное поглощение. $\forall_{ab}(a \subseteq b \to a \cup b = b)$. Имеются два приема: в первом антецедент содержится среди посылок; во втором - обрабатывается проверочным оператором, причем эта проверка жестко ограничивается по трудоемкости. Такое ограничение необходимо, чтобы на рассмотрение всевозможных пар операндов в длинных объединениях не затрачивалось слишком много времени. В отличие от безусловного поглощения, применявшегося на уровне 0, условное поглощение применяется на уровне 1. Для приема, использующего проверочный оператор, обращение продублировано уровнем 3. Это позволяет учесть дополнительные сведения, которые могут появиться в посылках при выводе следствий.

5. Дополнительные тождества типа поглощения. Приводится несколько полезных упрощающих тождеств:

   $\forall_{abc}(c \cup (a \cup c) \setminus b = c \cup (a \setminus b))$

   $\forall_{abc}(c \cup a \setminus (b \setminus c) = c \cup (a \setminus b))$

   $\forall_{ab}(b \cup (a \setminus b) = a \cup b)$

   $\forall_{abc}(c \cup a \setminus (b \cup c) = c \cup (a \setminus b))$

   $\forall_{abc}(b \subseteq c \to c \cup (a \setminus b) = a \cup c)$

   $\forall_{abcd}(c \subseteq a \ \& \ d = b \setminus c \to a \setminus b \cup c = a \setminus d)$

   Последний прием обобщает тождество $(a \setminus b) \cup (b \setminus c) = (a \setminus c)$, выполненное при $c \subseteq b \subseteq a$. Его второй антецедент - присвоение переменной d результата обработки разности $b \setminus c$ нормализатором "нормразность". Этот же нормализатор применяется и к заменяющему выражению. Уровень применения приема равен 3, в то время как предыдущие приемы применяются на уровнях 1 либо 0.

6. Свертка объединений. В обычном режиме вынесения за скобку общей части нескольких объединяемых выражений не происходит. Это делается лишь для задач, имеющих цель "свертка". Используются следующие тождества:

   $\forall_{abc}(a \cap b \cup a \cap c = a \cap (b \cup c))$

   $\forall_{abc}(a \cap c \cup a \setminus b = a \setminus (b \setminus c))$

   $\forall_{abc}(a \setminus b \cup c \setminus b = (a \cup c) \setminus b)$

Общая стандартизация утверждений

Для общей стандартизации утверждений с объединениями используются следующие приемы:

1. Развертка условия принадлежности объединению. $\forall_{abc}(a \in b \cup c \leftrightarrow a \in b \vee a \in c)$. Преобразование применяется "почти всегда", однако есть ряд ограничений.

   Прежде всего, если принадлежность является посылкой, то преобразование спровоцирует разбор случаев, а это далеко не всегда желательно. Поэтому данный прием вообще не применяется к корневым вхождения принадлежности в посылки. Однако, для посылок задач на доказательство сделано исключение - для них введена дополнительная версия данного приема, применяемая на уровне 4.

   Прием применяется на уровнях 0, 2 либо 3. Наибольший уровень 3 отведен для условий задач на доказательство. Это сделано, чтобы предоставить возможность в очевидных случаях обойтись без разбора случаев. Если принадлежность расположена в посылке под корневым отрицанием, то уровень равен 0 - здесь происходит разбиение посылки на два более простых утверждения. В остальных случаях уровень срабатывания равен 2.

   Далее проверяется отсутствие цели задачи, явно указывающей на режим "свертки" утверждений. Если редактируется ответ задачи на описание и a - неизвестная, то преобразование блокируется, чтобы не испортить стандартный вид ответа. Наконец, в процессе решения задачи на описание преобразование блокируется, если a содержит неизвестные, а b, c не содержат. Исключение делается лишь для случая, когда b либо c имеет вид "промежуток(...)" и выражение a - неизвестная, встречающаяся в других условиях. Тогда вынесение наружу условия принадлежности неизвестной промежутку может позволить сработать прочим приемам при объединении условий с этой неизвестной.

2. Развертка условия включения объединения. $\forall_{abc}(a \cup b \subseteq c \leftrightarrow (a \subseteq c \ \& \ b \subseteq c))$. Ограничения на срабатывание аналогичны предыдущему приему. Отличие состоит в том, что разбор случаев после применения данного приема инициируется, если принадлежность расположена под корневым отрицанием.

3. Развертка условия непересечения с объединением. $\forall_{abc}(\text{непересек}(c, a \cup b) \leftrightarrow \text{непересек}(a, c) \ \& \ \text{непересек}(b, c))$. Аналогично предыдущему приему.

4. Развертка условия пустоты объединения. $\forall_{ab}(a \cup b = \emptyset \leftrightarrow a = \emptyset \ \& \ b = \emptyset)$. Преобразование блокируется при наличии целей, указывающих на режим "свертки".

5. Усиление условия включения в объединение. $\forall_{abc}(\text{непересек}(a, b) \rightarrow (b \subseteq c \leftrightarrow b \subseteq (a \cup c)))$. Тождество применяется справа налево. Антецедент обрабатывается проверочным оператором. Уровень срабатывания приема равен 3.

6. Равенство множества объединению его с другим множеством. $\forall_{ab}(b \subseteq a \leftrightarrow a = a \cup b)$. Тождество применяется справа налево.

7. Усиление условия принадлежности объединению. $\forall_{abc}(\neg(a \in b) \rightarrow (a \in (b \cup c) \leftrightarrow a \in c))$. Уровень срабатывания равен 3. Потребность в данном приеме возникает, если развертка условия принадлежности была заблокирована.

Свертка известных утверждений при редактировании ответа

Несколько приемов упрощают входящие в ответ группы утверждений без неизвестных. Например, конъюнкция включений в одно и то же множество заменяется на включение объединения; дизъюнкция условий принадлежности - в условие принадлежности объединению, и т.п.

Усмотрение включения множества в объединение его с другим множеством

Простой прием с теоремой $\forall_{ab}(a \subseteq a \cup b)$.

Тождественные преобразования выражений с неизвестными

Несколько приемов, выполняющих группировку неизвестных членов. В каждом из них две "известных" переменных заносятся под общую операцию, играющую роль коэффициента при неизвестной. Ограничимся перечислением теорем этих приемов:

$\forall_{abc}(a \cap c \cup b \cap c = c \cap (a \cup b))$

$\forall_{abc}(b \setminus a \cup b \setminus c = b \setminus (a \cap c))$

$\forall_{abc}(a \cap c \cup a \setminus b = a \setminus (b \setminus c))$

$\forall_{abc}(a \setminus c \cup b \setminus c = (a \cup b) \setminus c)$

Эквивалентные преобразования условий с неизвестными

1. Разрешение относительно неизвестной условия включения множества в объединение двух других множеств. $\forall_{abc}(b \subseteq a \cup c \leftrightarrow b \setminus c \subseteq a)$. Выражение a содержит неизвестные; b, c - не содержат.

2. Два условия непересечения объединяются в условие непересечения с объединением. $\forall_{abc}(\text{непересек}(c, a \cup b) \leftrightarrow (\text{непересек}(a, c)\ \&\ \text{непересек}(b, c)))$. Заголовок приема - "заменаусловия(первыйтерм)". c - неизвестная; выражения a, b не содержат неизвестных. Этот и следующий приемы выполняют объединение в одно утверждение нескольких условий, уже явно разрешенных относительно неизвестной.

3. Два условия включения объединяются в условие включения объединения. $\forall_{abc}(a \cup b \subseteq c \leftrightarrow (a \subseteq c\ \&\ b \subseteq c))$. Здесь c - неизвестная; a, b известны.

4. Попытка представить неизвестную левую часть включения в виде объединения множеств. Если левая часть включения содержит неизвестные и представлена в виде объединения, то включение распадается на несколько более простых включений. Чтобы преобразовать левую часть к виду объединения, служит прием, обращающийся к нормализатору "видобъединение". Его теорема имеет вид $\forall_{cda}(a = d \rightarrow d \subseteq c \leftrightarrow a \subseteq c)$. Антецедент присваивает переменной a результат обработки выражения d указанным нормализатором. Заменяющее включение обрабатывается нормализаторами "уравнсодержится" и "нормлог", что в стандартных случаях завершает его явное разрешение относительно неизвестных. Прием применяется, если d отлично от переменной, не имеет заголовка "объединение" и содержит неизвестные. Проверяется, что выражение a либо имеет заголовок "объединение", либо короче выражения d.

5. Попытка представить неизвестную часть условия непересечения двух множеств в виде объединения. Аналогично предыдущему; теорема приема имеет вид: $\forall_{cda}(a = d \rightarrow$ непересек$(c, d) \leftrightarrow$ непересек$(c, a))$.

6. Попытка представить левую часть равенства пустому множеству в виде объединения. Аналогично предыдущему, но с более простой теоремой приема: $\forall_{ca}(a = c \rightarrow c = \emptyset \leftrightarrow a = \emptyset)$.

7. Усмотрение разбиения на два подмножества. Пусть условие задачи на описание имеет вид $B = x \cup y$, где x, y - выражения с неизвестными, B - известно. Если имеется также условие $x \subseteq A$, где A известно, причем усматривается, что y и A не пересекаются, то x равно $A \cap B$, y равно $B \setminus A$.

Кванторные свертки с объединением

Используются три приема кванторной свертки, основанные на следующих теоремах:

$\forall_{ab}(b - \text{set} \rightarrow (a \subseteq b) \leftrightarrow \exists_c(b = a \cup c \ \& \ c - \text{set}))$;

$\forall_{acd}(\forall_b(b \in c \ \vee \ b \in d \leftrightarrow b \in a) \leftrightarrow a = c \cup d)$;

$\forall_{abc}(c \subseteq a \cup b \leftrightarrow \forall_d(\neg(d \in a) \ \& \ d \in c \rightarrow d \in b))$

Попытка преобразования к виду объединения пересечений для упрощения выражения

Для упрощения выражений в алгебре множеств часто оказывается эффективным переход к виду объединения пересечений. Это - аналог дизъюнктивной нормальной формы алгебры логики. Преимущество обеих стандартных форм состоит в том, что любые два вхождения достижимы друг из друга с помощью малого числа переходов через символы операций. Соответственно, повышается вероятность обнаружить подходящее тождество, учитывающее взаимную зависимость таких вхождений. После упрощения в стандартной форме предпринимается попытка завершающей компактной переформулировки выражения (группировки с вынесением за скобки и т.п.).

Приемы, выполняющие указанные действия, созданы для каждого из символов теоретико-множественных операций. Если заголовком преобразуемого выражения служит объединение, то теорема приема имеет вид: $\forall_{abc}(c = a \cup b \rightarrow a \cup b = c)$. Антецедент присваивает переменной c результат упрощений. Чтобы преобразовать выражение к виду объединения пересечений и упростить его в данном виде, служит нормализатор "стандобъединение". Завершающие упрощения обеспечиваются последовательным применением нормализаторов "группмножество" и "свертка".

Уровень срабатывания приема равен 3. Фильтр "не(контекст(операнд(x4 теквхожд)символ(x4 объединение пересечение разность прямоепроизведение прообраз образ)))" позволяет упрощать только "максимальные" подвыражения и таким образом исключить повторы. Указатель "попытка(стандобъединение теквхожд)" вводит комментарий, блокирующий повторное упрощение при неудаче. Указатель "замечание(стандобъединение результат)" вводит комментарий, блокирующий повторное упрощение результата преобразований.

Нормализатор общей стандартизации "нормобъединение"

Нормализаторы общей стандартизации выражений состоят из приемов, выполняющих такие преобразования, которые бесспорно следует применять во всех случаях. Возможны редкие исключения, которые либо явно указываются в фильтрах приемов, либо реализуются косвенно - путем блокировки обращений к нормализатору. Теоремы приемов нормализатора очень просты, и мы приводим их без замечаний:

1. Устранение вложенных объединений;
2. Лексикографическое упорядочение операндов;
3. Объединение с пустым множеством. $\forall_a (a \cup \emptyset = a)$;
4. Одинаковые операнды. $\forall_a (a \cup a = a)$;
5. Объединение конечных списков. $\forall_{ab} (\{;a\} \cup \{;b\} = \{;a;b\})$;
6. Объединение с конечным списком. $\forall_{abc} (b \in a \to a \cup \{b;c\} = a \cup \{;c\})$;
7. Безусловное поглощение. $\forall_{ab} (a \cup a \cap b = a)$; $\forall_{ab} (a \cup a \setminus b = a)$;
8. Условное поглощение. $\forall_{ab} (a \subseteq b \to a \cup b = b)$;
9. Дополнительные тождества типа поглощения.
   $\forall_{abc} (c \cup b \cap (a \cup c) = c \cup a \cap b)$
   $\forall_{abc} (b \subseteq c \to c \cup a \setminus b = a \cup c)$
   $\forall_{abc} (c \cup a \setminus (b \cup c) = c \cup a \setminus b)$
   $\forall_{ab} (b \cup a \setminus b = a \cup b)$
   $\forall_{abc} (c \cup a \setminus (b \setminus c) = c \cup a \setminus b)$
   $\forall_{abc} (c \cup (a \cup c) \underset{-}{=} c \cup a \setminus b)$
10. Объединение промежутка и конечного списка, содержащего его конечную точку.
    $\forall_{abcde} (\{b;d\} \cup [a,b] = \{;d\} \cup [a,b])$
    $\forall_{abcde} (\{a;d\} \cup [a,b] = \{;d\} \cup [a,b])$

    В левой части тождества условия на концы промежутка произвольные; в правой - соответствующий добавляемой точке конец включается в промежуток.
11. Объединение двух подсемейств, полученных отбрасыванием различных элементов.
    $\forall_{abAP} (\neg(a - b = 0) \to \bigcup_{i,\neg(i=a),P(i)} A(i) \cup \bigcup_{j,\neg(j=b),P(j)} A(j) = \bigcup_{i,P(i)} A(i))$
12. Объединение промежутков.
    $\forall_{abcdef} (0 \leq c - a \ \&\ 0 \leq b - d \to [a,b] \cup [c,d] = [a,b])$

    У промежутка $[c,d]$ условия на концы произвольны; промежуток $[a,b]$ включает в себя свои концы. Фильтр "или(равно(x1 x3)равно(x2 x4))" означает, что в действительности попытка применения приема будет иметь место лишь для промежутков, у которых один из концов - общий.

Нормализатор сокращенной записи "упрощобъединение"

При завершающем редактировании ответов задач обычно используется режим компактной переформулировки термов. Такая переформулировка на основном этапе решения часто нежелательна, так как затрудняет применение текущих преобразований. Например, при упрощении выражения выгодно степень произведения преобразовать к виду произведения степеней. Это упрощает умножение и деление степеней с одинаковыми основаниями. Однако, после выполнения "основного" упрощения обычно происходит обратная свертка сомножителей под общий показатель степени.

Для выполнения сокращенной переформулировки используется процедура "свертка". В свою очередь, она прибегает к целой системе нормализаторов сокращенной перезаписи, соответствующих различным логическим символам. Названия этих нормализаторов начинаются с приставки "упрощ" - "упрощобъединение", "упрощумножение", и т.п. Чтобы найти название нормализатора, соответствующего заданному символу, служит справочник "нормупростить".

Справочники, определяющие формат оператора "упрощобъединение", аналогичны справочникам для оператора "нормобъединение". Отличие состоит лишь в том, что вместо справочника "нормализатор" используется справочник "нормупростить". Обращаться из нормализаторов сокращенной перезаписи к нормализаторам общей стандартизации следует с некоторой осторожностью, так как эти обращения могут привести к нежелательной развертке свернутых выражений.

Приемы оператора "упрощобъединение" таковы:

1. Устранение вложенных объединений и лексикографическое упорядочение операндов;
2. Объединение прообразов. $\forall_{abc}(\text{прообраз}(c,a) \cup \text{прообраз}(c,b) = \text{прообраз}(c, a \cup b))$;
3. Объединение образов. $\forall_{abc}(\text{образ}(c,a) \cup \text{образ}(c,b) = \text{образ}(c, a \cup b))$;
4. Объединение разностей либо пересечений.

   $\forall_{abc}(b \setminus a \cup b \setminus c = b \setminus (a \cap c))$

   $\forall_{abcde}(a \setminus (b \cup c) \cup d \setminus (c \cup e) = (a \setminus b \cup d \setminus e) \setminus c)$

   $\forall_{abcde}((b \cap e) \setminus a \cup (d \cap e) \setminus c = e \cap (b \setminus a \cup d \setminus c))$

   $\forall_{abcd}(d \setminus a \cup (c \cap d) \setminus b = d \setminus (a \setminus (c \setminus b)))$

   Во второй теореме b, e могут вырождаться в пустое множество; в теретьей - a, c; в четвертой - b. Эти параметры введены в качестве обобщающих коэффициентов.
5. Объединение прямых произведений.

   $\forall_{abc}((b \times a) \cup (b \times c) = b \times (a \cup c))$

   $\forall_{abc}((a \times c) \cup (b \times c) = (a \cup b) \times c)$
6. Объединение объединений семейств.

   $\forall_{fgh}(\bigcup_{x,h(x)} f(x) \cup \bigcup_{x,h(x)} g(x) = \bigcup_{x,h(x)}(f(x) \cup g(x)))$
7. Усмотрение симметрической разности.

   $\forall_{ab}(a \setminus b \cup b \setminus a = a \triangle b)$

Нормализатор "уравнобъединение"

Этот нормализатор служит для преобразований, осуществляющих приведение подобных членов с неизвестными. Его приемы таковы:

1. Объединение пересечений. $\forall_{abc}(a \cap c \cup b \cap c = c \cap (a \cup b))$. Выражение c содержит неизвестные; a, b известны. К заменяющему выражению применяются нормализаторы "нормпересечение", "уравнпересечение". К подвыражению $a \cup b$ применяется нормализатор "нормобъединение". Аналогичными нормализаторами снабжены и остальные приемы пакета.
2. Объединение с пересечением. $\forall_{abc}(c \subseteq b \rightarrow c \cup a \cap b = b \cap (a \cup c))$. Выражение b не известно; a, c известны. Антцедент обрабатывается проверочным оператором.
3. Объединение прообразов. $\forall_{abc}(\text{прообраз}(c, a) \cup \text{прообраз}(c, b) = \text{прообраз}(c, a \cup b))$. Выражение c не известно; a, b известны.
4. Объединение образов. $\forall_{abc}(\text{образ}(c, a) \cup \text{образ}(c, b) = \text{образ}(c, a \cup b))$. Выражение c не известно; a, b - известны.
5. Объединение прямых произведений. $\forall_{abc}((b \times a) \cup (b \times c) = b \times (a \cup c))$. Выражение b не известно; a, c известны. Аналогичный прием - для умножения слева.
6. Объединение разностей.

   $\forall_{abc}(a \setminus c \cup b \setminus c = (a \cup b) \setminus c)$

   $\forall_{abc}(b \setminus a \cup b \setminus c = b \setminus (a \cap c))$
7. Объединение разности и пересечения. $\forall_{abc}(a \cap c \cup a \setminus b = a \setminus (b \setminus c))$.
8. Объединение с разностью.

   $\forall_{abc}(a \subseteq c \rightarrow a \cup c \setminus b = c \setminus (b \setminus a))$

   $\forall_{abc}(\text{непересек}(b, c) \rightarrow b \cup a \setminus c = (a \cup b) \setminus c)$

Нормализатор "видобъединение"

Этот нормализатор преобразует выражения к виду объединения. В отличие от рассматриваемого ниже нормализатора "стандобъединение", он не выполняет каких-либо специальных упрощений и применяется редко. Кроме несложных преобразований типа дистрибутивности, нормализатор обращается к подготавливающим возможность таких преобразований пакетам. Например, если выражение имеет вид $a \cap b$, то предпринимается рекурсивное обращение к этому же пакету для приведения a к виду объединения.

Нормализатор "стандобъединение"

Нормализатор преобразует выражения к виду объединения таких членов, которые далее уже не представляются естественным образом в виде объединения. Для этих членов предпринимается некоторая дополнительная стандартизация. Оставаясь в рамках данной стандартизации, нормализатор далее выполняет упрощение выражения. Заметим, что в формате нормализатора отсутствует символ "корень", т.е. его тождества применяются и к внутренним частям преобразуемого выражения. Справочник "стандформа" позволяет по операции "объединение" определять название нормализатора, преобразующей к стандартной форме с этой корневой операцией. Справочник "станд" перечисляет шаблоны тех подвыражений, которые становятся запрещенными для выражений, приведенных к стандартной форме. Фактически это левые части соответствующих тождеств, используемых в процессе приведения.

Перечислим приемы нормализатора.

1. Приведение к стандартной форме. Используются следующие тождества:

   $\forall_{abc}(c \cap (a \cup b) = a \cap c \cup b \cap c)$

   $\forall_{abc}(c \cap (b \setminus a) = (b \cap c) \setminus a)$

   $\forall_{abc}((b \setminus c) \setminus a = b \setminus (a \cup c))$

   $\forall_{abc}((a \cup b) \setminus c = a \setminus c \cup b \setminus c)$

   $\forall_{bcef}(c \setminus (e \cap f \cup b) = c \setminus (b \cup e) \cup c \setminus (b \cup f))$

   $\forall_{bcef}(c \setminus (e \setminus f \cup b) = c \setminus (b \cup e) \cup (c \cap f) \setminus b)$

   $\forall_{abc}((a \cup b) \times c = (a \times c) \cup (b \times c))$

   $\forall_{abc}(b \times (a \cup c) = (b \times a) \cup (b \times c))$

   $\forall_{abc}(a \times (c \setminus b) = (a \times c) \setminus (a \times b))$

   $\forall_{abc}((c \setminus b) \times a = (c \times a) \setminus (b \times a))$

   $\forall_{abcd}((a \cap b) \times (c \cap d) = (a \times c) \cap (b \times d))$

   $\forall_{abc}(\text{образ}(c, a \cup b) = \text{образ}(c, a) \cup \text{образ}(c, b))$

   $\forall_{abc}(\text{прообраз}(a, c) \setminus \text{прообраз}(a, b) = \text{прообраз}(a, c \setminus b))$

   $\forall_{abc}(\text{прообраз}(c, a) \cup \text{прообраз}(c, b) = \text{прообраз}(c, a \cup b))$

   $\forall_{abc}(\text{прообраз}(c, a) \cap \text{прообраз}(c, b) = \text{прообраз}(c, a \cap b))$

   $\forall_{fgh}(\bigcup_{x,h(x)}(f(x) \cup g(x)) = \bigcup_{x,h(x)} f(x) \cup \bigcup_{x,h(x)} g(x))$

   $\forall_{fgh}(\neg(\mathrm{set}_x(h(x)) = \emptyset) \to \bigcap_{x,h(x)}(f(x) \setminus g(x)) = \bigcap_{x,h(x)} f(x) \setminus \bigcup_{x,h(x)} g(x))$

   $\forall_{fgh}(\neg(\mathrm{set}_x(h(x)) = \emptyset) \to \bigcap_{x,h(x)}(f(x) \cap g(x)) = \bigcap_{x,h(x)} f(x) \cap \bigcap_{x,h(x)} g(x))$

   Все эти тождества применяются слева направо. Их уровень срабатывания равен 1. В отсутствии более сложных операций, выражение приводится к виду объединения таких членов, каждый из которых является либо пересечением переменных, либо разностью, имеющей своей левой частью пересечение, а правой - объединение переменных.

2. Простейшие упрощающие тождества. Безусловные тождества, имеющие не более двух переменных, применяются на уровне 1, остальные - на уровне 2. На первом уровне упрощающие преобразования могут перемежаться с преобразованиями, приводящими к стандартной форме.

   $\forall_a(a \cap \emptyset = \emptyset)$

   $\forall_a(a \cup \emptyset = a)$

   $\forall_a(a \cup a = a)$

   $\forall_a(a \cap a = a)$

$\forall_{ab}(a \cup a \cap b = a)$

$\forall_{ab}(a \cup a \setminus b = a)$

$\forall_{abc}(b \cup (a \cap b) \setminus c = b)$

$\forall_{abc}(b \setminus a \cup (b \cap c) \setminus a = b \setminus a)$

$\forall_{abc}(b \setminus c \cup b \setminus (a \cup c) = b \setminus c)$

$\forall_{ab}(a \subseteq b \rightarrow a \cup b = b)$

$\forall_{ab}(b \subseteq a \rightarrow a \cap b = b)$

$\forall_{ab}(\text{непересек}(a, b) \rightarrow a \cap b = \emptyset)$

$\forall_{a}(\emptyset \setminus a = \emptyset)$

$\forall_{a}(a \setminus \emptyset = a)$

$\forall_{a}(a \setminus a = \emptyset)$

$\forall_{ab}(a \subseteq b \rightarrow a \setminus b = \emptyset)$

$\forall_{ab}((a \cap b) \setminus b = \emptyset)$

$\forall_{ab}(a \setminus (a \cup b) = \emptyset)$

$\forall_{abc}((a \cap b) \setminus (a \cup c) = \emptyset)$

$\forall_{ab}(\text{непересек}(a, b) \rightarrow a \setminus b = a)$

$\forall_{abc}(\text{непересек}(b, c \setminus a) \rightarrow c \setminus (a \cup b) = c \setminus a)$

$\{;\text{пустоеслово}\} = \emptyset$

$\forall_{a}(a \times \emptyset = \emptyset)$

$\forall_{a}(\emptyset \times a = \emptyset)$

$\forall_{a}(\text{образ}(a, \emptyset) = \emptyset)$

$\forall_{a}(\text{прообраз}(a, \emptyset) = \emptyset)$

Пустые слова могут возникать при использовании приводимых ниже тождеств, преобразующих конечные списки.

Перечислим более сложные упрощающие тождества пакета. Уровень срабатывания их равен 2.

3. Склеивание двух членов объединения.

   $\forall_{abc}(a - \text{set} \ \& \ \text{непересек}(b, c) \rightarrow a \setminus b \cup a \setminus c = a)$

   $\forall_{abc}(a - \text{set} \ \& \ b \subseteq c \rightarrow a \cap c \cup a \setminus b = a)$

   $\forall_{ab}(a - \text{set} \rightarrow a \cap b \cup a \setminus b = a)$

   $\forall_{abc}(b \setminus (a \cup c) \cup (b \cap c) \setminus a = b \setminus a)$

   $\forall_{abc}(a \subseteq c \ \& \ c \subseteq b \rightarrow b \setminus c \cup c \setminus a = b \setminus a)$

4. Упрощение для двух членов объединения.

   $\forall_{abc}(b \subseteq c \rightarrow c \cup a \setminus b = a \cup c)$

   $\forall_{abc}(c \cup a \setminus (b \cup c) = c \cup a \setminus b)$

   $\forall_{ab}(b \cup a \setminus b = a \cup b)$

$\forall_{abc}(a \cap c \cup (a \cap b) \setminus c = a \cap b \cup a \cap c)$

$\forall_{abc}(b \setminus (a \cup c) \cup c \setminus a = b \setminus a \cup c \setminus a)$

$\forall_{abc}(b \cap c \cup b \setminus (a \cup c) = b \cap c \cup b \setminus a)$

$\forall_{abc}(c \subseteq b \rightarrow a \cap b \cup c \setminus a = c \cup a \cap b)$

$\forall_{abc}(a \cap b \cap c \cup b \setminus a = b \cap c \cup b \setminus a)$

$\forall_{abc}(\text{непересек}(b, c) \rightarrow a \setminus c \cup b \setminus a = b \cup a \setminus c)$

$\forall_{abc}(b \subseteq c \rightarrow a \cap b \cup c \setminus a = b \cup c \setminus a)$

$\forall_{abc}(a \setminus b \cup (a \cap b) \setminus c = a \setminus b \cup a \setminus c)$

5. Поглощение для трех членов объединения.

   $\forall_{abc}(a \setminus b \cup a \setminus c \cup b \setminus c = a \setminus b \cup b \setminus c)$

   $\forall_{abc}(a \cap c \cup b \cap c \cup a \setminus b = b \cap c \cup a \setminus b)$

6. Склеивание трех членов объединения.

   $\forall_{abc}(a \cap c \cup b \setminus a \cup b \setminus c = b \cup a \cap c)$

   $\forall_{abc}(a \cap c \cup a \setminus b \cup b \setminus c = a \cup b \setminus c)$

7. Тождества с прямым произведением.

   $\forall_{abc}((a \times b \cap c) \setminus (a \times c) = \emptyset)$

   $\forall_{abc}(\text{непересек}(b, c) \rightarrow (a \times b) \setminus (a \times c) = a \times b)$

   $\forall_{abc}((b \cap c \times a) \setminus (c \times a) = \emptyset)$

   $\forall_{abc}(\text{непересек}(b, c) \rightarrow (b \times a) \setminus (c \times a) = b \times a)$

   $\forall_{abc}((a \times b) \cup (a \times b \cap c) = a \times b)$

   $\forall_{abc}((b \times a) \cup (b \cap c \times a) = b \times a)$

8. Тождества с конечными списками.

   В приводимых ниже тождествах запись $\{; a; b\}$ обозначает конечное множество, элементы которого перечисляются конкатенацией наборов a, b; запись $\{a; b\}$ - конечное множество, элементы которого перечисляются добавлением элемента a к набору b. Посредством $\{; a\}$ обозначается конечное множество, элементы которого перечисляются набором a.

   $\forall_{abc}(\neg(b \in a) \rightarrow a \cap \{b; c\} = a \cap \{; c\})$

   $\forall_{ab}(\{; a\} \cup \{; b\} = \{; a; b\})$

   $\forall_{abc}(\neg(b \in a) \rightarrow a \setminus \{b; c\} = a \setminus \{; c\})$

   $\forall_{abc}(b \in a \rightarrow a \cup \{b; c\} = a \cup \{; c\})$

   $\forall_{ab}(\{a, a; b\} = \{a; b\})$

   $\forall_{a}(\{; a; a\} = \{; a\})$

   $\forall_{abc}(a \cap \{; b\} \cup \{; b; c\} = \{; b; c\})$

   $\forall_{abc}(a \setminus \{; b\} \cup \{; b; c\} = a \cup \{; b; c\})$

   $\forall_{abc}(\{; b\} \setminus a \cup \{; b; c\} = \{; b; c\})$

   $\forall_{abc}(\{; a\} \setminus (c \cup \{; a; b\}) = \emptyset)$

$\forall_{abc}(a \cap \{;b\} \cup a \cap \{;c\} = a \cap \{;b;c\})$

$\forall_{abc}(\{;b\} \setminus a \cup \{;c\} \setminus a = \{;b;c\} \setminus a)$

$\forall_{abc}(a \setminus \{;b;c\} \cup \{;c\} = a \setminus \{;b\} \cup \{;c\})$

$\forall_{abc}((a \cap \{;b\}) \setminus \{;b;c\} = \emptyset)$

$\forall_{abc}(a \cap \{;b;c\} \cup \{;b\} = a \cap \{;c\} \cup \{;b\})$

$\forall_{ab}(\{;a\} \cap \{;a;b\} = \{;a\})$

$\forall_{abc}(a \setminus \{;b\} \cup a \setminus \{;b;c\} = a \setminus \{;b\})$

$\forall_{abc}(\{;a;b\} \setminus c \cup \{;a\} = \{;b\} \setminus c \cup \{;a\})$

$\forall_{abc}(\{;a;b\} \setminus (c \cup \{;a\}) = \{;b\} \setminus (c \cup \{;a\}))$

$\forall_{abc}(\{;a;c\} \setminus \{b;c\} = \{;a\} \setminus \{b;c\})$

$\forall_{abc}(\{;a\} \setminus \{;b;c\} \cup \{;a;c\} \setminus \{;b\} = \{;a;c\} \setminus \{;b\})$

$\forall_{abc}(\{b;a\} \setminus \{b;c\} = \{;a\} \setminus \{b;c\})$

$\forall_{abc}(b \in a \rightarrow \{b;c\} \setminus a = \{;c\} \setminus a)$

Столь большое количество тождеств объясняется очень просто - они были сгенерированы автоматически при экспериментах с логическим выводом в базе теорем и оставлены в решателе. Большинство из них поглощается простейшими условными тождествами. Однако, наличие готового "безусловного" шаблона для частной ситуации иногда ускоряет вычисления.

Это же замечание относится и к последующим пунктам.

9. Тождества с прямыми произведениями и конечными списками.

   $\forall_{abc}((a \times \{;b\}) \cup (a \times \{;c\}) = a \times \{;b;c\})$

   $\forall_{abc}((\{;b\} \times a) \cup (\{;c\} \times a) = \{;b;c\} \times a)$

   $\forall_{abc}((a \times \{;b\}) \setminus (a \times \{;b;c\}) = \emptyset)$

   $\forall_{abc}((\{;b\} \times a) \setminus (\{;b;c\} \times a) = \emptyset)$

   $\forall_{abcd}(c \in b \rightarrow (a \times b) \cup (a \times \{c;d\}) = (a \times b) \cup (a \times \{;d\}))$

   $\forall_{abcd}(c \in b \rightarrow (b \times a) \cup (\{c;d\} \times a) = (b \times a) \cup (\{;d\} \times a))$

10. Тождества для образа.

    $\forall_{abc}(\text{образ}(a,b) \cup \text{образ}(a,b \cap c) = \text{образ}(a,b))$

    $\forall_{abc}(\text{образ}(a,c) \cup \text{образ}(a,b \setminus c) = \text{образ}(a,b) \cup \text{образ}(a,c))$

    $\forall_{abc}(\text{образ}(a,b) \cup \text{образ}(a,b \setminus c) = \text{образ}(a,b))$

    $\forall_{abc}(\text{образ}(a,b \cap c) \setminus \text{образ}(a,b) = \emptyset)$

    $\forall_{abc}(\text{образ}(a,b \setminus c) \setminus (\text{образ}(a,b) \cup \text{образ}(a,c)) = \emptyset)$

    $\forall_{abc}(\text{образ}(a,b \setminus c) \setminus \text{образ}(a,b) = \emptyset)$

    $\forall_{abc}(\text{образ}(a,b \setminus c) \setminus \text{образ}(a,c) \cup \text{образ}(a,b) = \text{образ}(a,b))$

11. Тождества для образов и конечных списков.

    $\forall_{abc}(\text{образ}(a,\{;b\} \setminus \{;c\}) \cup \text{образ}(a,\{;b;c\}) = \text{образ}(a,\{;b;c\}))$

    $\forall_{abc}(\text{образ}(a,\{;b\} \setminus \{;c\}) \setminus \text{образ}(a,\{;b;c\}) = \emptyset)$

$\forall_{abc}(\text{образ}(a, \{;b\}) \setminus \text{образ}(a, \{;b;c\}) = \emptyset)$

$\forall_{abc}(\text{образ}(a, \{;b\}) \cup \text{образ}(a, \{;c\}) = \text{образ}(a, \{;b;c\}))$

4.6 Приемы символа "пересечение"

Приемы этого раздела мало чем отличаются от приемов предыдущего. Они основаны на столь же простых теоремах и имеют мало управляющей логики. Для пересечения не введен аналог нормализатора "стандобъединение" - вполне достаточно оказалось использовать стандартную форму типа "объединение пересечений".

Общая стандартизация выражений

Для общей стандартизации выражений, имеющих заголовок "пересечение", используются следующие приемы:

1. Пересечение с пустым множеством. $\forall_a(a \cap \emptyset = \emptyset)$
2. Пересечение одинаковых множеств. $\forall_a(a \cap a = a)$
3. Пересечение непересекающихся множеств. $\forall_{ab}(\text{непересек}(a, b) \to a \cap b = \emptyset)$
4. Пересечение с разностью. $\forall_{abc}(c \cap (b \setminus a) = (b \cap c) \setminus a)$
5. Безусловное поглощение. $\forall_{ab}(a \cap (a \cup b) = a)$
6. Условное поглощение. $\forall_{ab}(b \subseteq a \to a \cap b = b)$. Предусмотрены две версии приема. В одной из них переменные a, b идентифицируются с отдельными операндами пересечения (возможно, имеющего более двух операндов). Включение проверяется пакетным оператором. В другой версии - включение идентифицируется с некоторым утверждением из контекста, а его части образуют подмножества операндов пересечения. Чтобы вторая версия не дублировала первой, требуется, чтобы левая часть включения имела заголовок "пересечение".
7. Дополнительные тождества типа поглощения.

   $\forall_{abc}(c \cup b \cap (a \cup c) = c \cup a \cap b)$

   $\forall_{abc}(c \cap (b \cup a \cap c) = c \cap (a \cup b))$

   $\forall_{abc}(\text{непересек}(b, c) \to c \cap (a \cup b) = a \cap c)$

   $\forall_{cdfa}(d \subseteq c \ \& \ a = c \cap f \to c \cap (d \cup f) = d \cup a)$

   В последнем приеме переменной a присваивается результат обработки выражения $c \cap f$ нормализатором "нормпересечение" и проверяется, что он короче данного выражения.

Общая стандартизация утверждений

Для упрощения утверждений, содержащих символ "пересечение", используются следующие приемы:

1. Пустота пересечения двух множеств. $\forall_{ab}(a \cap b = \emptyset \leftrightarrow \text{непересек}(a, b))$

2. Условие непересечения пересечения двух множеств с одним из них.

   $\forall_{ab}$(непересек$(a, b) \leftrightarrow$ непересек$(a, a \cap b)$). Эквивалентность применяется справа налево.

3. Условие равенства пересечения двух множеств одному из них.

   $\forall_{ab}(a \subseteq b \leftrightarrow a = a \cap b)$. Эквивалентность применяется справа налево.

4. Условие непересечения пересечения двух множеств с подмножеством одного из них.

   $\forall_{abc}(c \subseteq a \rightarrow$ непересек$(b, c) \leftrightarrow$ непересек$(c, a \cap b)$). Эквивалентность применяется справа налево.

5. Включение в пересечение. $\forall_{abc}(c \subseteq a \cap b \leftrightarrow c \subseteq a \ \&\ c \subseteq b)$. Фильтры приема аналогичны фильтрам приема развертки включения объединения.

6. Принадлежность пересечению. $\forall_{abc}(a \in b \cap c \leftrightarrow a \in b \ \&\ a \in c)$. Фильтры аналогичны фильтрам приема развертки принадлежности объединению.

Свертка утверждений при редактировании ответа

Чтобы объединить при редактировании ответа задачи на описание несколько утверждений, не содержащих неизвестных, используются следующие приемы:

1. Свертка конъюнкции включений. $\forall_{abc}(c \subseteq a \cap b \leftrightarrow c \subseteq a \ \&\ c \subseteq b)$. Эквивалентность применяется справа налево.

2. Свертка двух условий включения. Теорема та же, что в предыдущем приеме, но объединяются не конъюнктивные члены, а независимые условия задачи.

3. Свертка конъюнкции двух принадлежностей в принадлежность пересечению. $\forall_{abc}(a \in b \cap c \leftrightarrow a \in b \ \&\ a \in c)$.

4. Свертка двух условий принадлежности в принадлежность пересечению.

Усмотрение включения пересечения двух множеств в одно из них

$\forall_{ab}(a \cap b \subseteq a)$

Кванторные свертки с участием пересечения множеств

Для перехода к бескванторным утверждениям применяются (при отсутствии блокирующего комментария "кванторнаясвертка") следующие эквивалентности:

$\forall_{ab}(b - \mathrm{set} \rightarrow b \subseteq a \leftrightarrow \exists_c(b = a \cap c \ \&\ c - \mathrm{set}))$

$\forall_{abc}$(непересек$(c, a \cap b) \leftrightarrow \forall_d(d \in a \ \&\ d \in b \rightarrow \neg(d \in c))$)

$\forall_{acd}(\forall_b(b \in c \ \&\ b \in d \leftrightarrow b \in a) \leftrightarrow a = c \cap d)$

$\forall_{abc}(a \cap b \subseteq c \leftrightarrow \forall_d(d \in a \ \&\ d \in b \rightarrow d \in c))$

Группировка операндов с неизвестными

Для приведения подобных членов с неизвестными в ситуации, когда роль "сложения" играет пересечение, а роль "умножения" - объединение, используется следующий прием: $\forall_{abc}((a \cup b) \cap (a \cup c) = a \cup (b \cap c))$. Здесь a содержит неизвестные; b, c известны.

Эквивалентные преобразования условий с неизвестными

Для явного разрешения относительно неизвестных условий задач на описание, содержащих пересечение, используются следующие приемы:

1. Разрешение относительно неизвестной условия непересечения множества с пересечением двух других множеств.

   $\forall_{abc}(\text{непересек}(c, a \cap b) \leftrightarrow \text{непересек}(a, b \cap c))$.

   Эквивалентность применяется слева направо. a - все операнды пересечения, содержащие неизвестные. Выражения b, c известны.

2. Разрешение относительно неизвестной условия включения пересечения.

   $\forall_{abc}(\text{непересек}(c, a \setminus b) \leftrightarrow a \cap c \subseteq b)$.

   Эквивалентность применяется справа налево. c - все операнды с неизвестными; a, b известны.

3. Группировка нескольких разрешенных относительно неизвестной условий в одно.

   $\forall_{abc}(a \in b \cap c \leftrightarrow a \in b \ \& \ a \in c)$

   $\forall_{abc}(c \subseteq a \cap b \leftrightarrow c \subseteq a \ \& \ c \subseteq b)$

   В первом случае a, а во втором c - неизвестная. Прочие переменные идентифицируются с известными выражениями. При наличии цели "пример" первое преобразование вообще не применяется, а второе - применяется на более высоком уровне. Если a - числовая неизвестная, встречающаяся в уравнениях, то первый прием блокируется.

4. Группировка неизвестных в общем подвыражении. Если выражения с неизвестными расположены в разных частях двуместного отношения, то иногда бывает полезно переформулировать это отношение, сгруппировав неизвестные в одном выражении. Для этой цели используются следующие приемы:

   $\forall_{ab}(a \cap b = \emptyset \leftrightarrow \text{непересек}(a, b)$

   $\forall_{abc}(\text{непересек}(c, a \cap b) \leftrightarrow \text{непересек}(a, b \cap c))$

   В обоих случаях замена выполняется справа налево. Оба выражения a, b содержат неизвестные; c - известно. Заменяющие утверждения обрабатываются нормализаторами уравнений (соответственно, "уравнмножество" и "уравннепересек"). Предварительно выражение $a \cap b$ обрабатывается нормализаторами "нормпересечение", "уравнпересечение". Таким образом, обычно прием доводит процесс разрешения утверждения относительно неизвестной до конца.

Декомпозиция условий задач на доказательство и посылок

Для преобразования условий задач на доказательство и посылок используются следующие эквивалентности декомпозиционного типа:

$\forall_{bcef}(\text{непересек}(c,(e\cup f)\cap b)\leftrightarrow\text{непересек}(b\cap e,c)\ \&\ \text{непересек}(b\cap f,c))$

$\forall_{bcef}(c\subseteq e\cap f\cup b\leftrightarrow c\subseteq b\cup e\ \&\ c\subseteq b\cup f)$

$\forall_{bcef}((e\cup f)\cap b\subseteq c\leftrightarrow b\cap e\subseteq c\ \&\ b\cap f\subseteq c)$

Попытка преобразования к виду объединения пересечений

Чтобы упрощать выражения с заголовком "пересечение", имеется прием, предпринимающий попытку применения стандартной формы "стандобъединение". Он аналогичен уже рассмотренному выше приему для заголовка "объединение". Теорема приема имеет вид: $\forall_{abc}(c=a\cap b\rightarrow a\cap b=c)$. Переменной c присваивается результат обработки пересечения нормализаторами "стандобъединение", "группмножество", "свертка".

Вывод следствий

Если посылка задачи на доказательство либо на исследование имеет вид $a=b\cap c$, то из нее выводится следствие $a\subseteq b$. Так как b идентифицируется с произвольным операндом пересечения, то в действительности порождаются следствия указанного вида для всех операндов.

Нормализаторы

С символом "пересечение" связаны нормализаторы, аналогичные нормализаторам символа "объединение". Во- первых, это нормализатор общей стандартизации "нормпересечение". Во - вторых, нормализатор приведения подобных членов с неизвестными "уравнпересечение" (он относится только к выражениям, имеющим заголовок "пересечение"). В - третьих, нормализатор сокращенной записи "упрощпересечение". Запись на ГЕНОЛОГе приемов перечисленных нормализаторов не содержит какихлибо новых моментов, и мы их здесь не приводим.

4.7 Приемы символа "разность"

Приемы этого раздела аналогичны приемам двух предыдущих разделов. Они завершают рассмотрение основных операций алгебры множеств.

Общая стандартизация выражений

Для приведения к стандартному виду выражений, имеющих заголовок "разность", используются следующие приемы:

1. Вычитание пустого множества. $\forall_a(a\setminus\emptyset=a)$;
2. Вычитание из пустого множества. $\forall_a(\emptyset\setminus a=\emptyset)$;
3. Разность одинаковых множеств. $\forall_a(a\setminus a=\emptyset)$;

4. Пустота разности множества и надмножества.

   $\forall_{ab}(a \setminus (a \cup b) = \emptyset)$

   $\forall_{ab}((a \cap b) \setminus b = \emptyset)$

   $\forall_{ab}(a \subseteq b \to a \setminus b = \emptyset)$

   Первые два приема срабатывают на уровне 0, последний - на уровнях 1 и 3 (второе значение - для контроля изменения контекста).

5. Вычитание непересекающихся множеств.

   $\forall_{ab}(b \setminus (a \setminus b) = b)$

   $\forall_{ab}(\text{непересек}(a, b) \to a \setminus b = a)$

   $\forall_{abc}(\text{непересек}(a, b) \to a \setminus (b \cup c) = a \setminus c)$

   Первый прием срабатывает на уровне 0, последние два - на уровнях 1 и 3.

6. Вычитание из разности. $\forall_{abc}(b \setminus (a \cup c) = (b \setminus c) \setminus a)$. Тождество применяется справа налево.

7. Вычитание разности.

   $\forall_{ac}(a \setminus (a \setminus c) = a \cap c)$

   $\forall_{bde}((b \cap e) \setminus (b \setminus d) = b \cap d \cap e)$

   $\forall_{abc}(c \subseteq a \to a \setminus b \cup c = a \setminus (b \setminus c))$

   Первые два приема применяются на уровне 0; последний - на уровне 2. Его тождество ориентировано справа налево, причем заблокирована замена, когда a содержит неизвестные, а b, c - не содержат.

8. Вычитание из объединения. $\forall_{abc}(\text{непересек}(a, c) \to (a \cup b) \setminus c = a \cup b \setminus c)$. Уровень срабатывания равен 2; заблокирован случай, когда c не известно, а a, b известны.

9. Вычитание объединения. $\forall_{bcd}(b \setminus (d \cup b \setminus c) = (b \cap c) \setminus d)$.

10. Вычитание множества из объединения его с другим множеством. $\forall_{ab}((a \cup b) \setminus b = a \setminus b)$

11. Вычитание из множества его пересечения с другим множеством. $\forall_{ab}(b \setminus (a \cap b) = b \setminus a)$

12. Использование условия непересечения множеств для упрощения разности.

    $\forall_{abc}(\text{непересек}(b, c) \to b \setminus (a \setminus c) = b \setminus a)$

    $\forall_{abc}(\text{непересек}(b, c) \to b \setminus (a \cup c) = b \setminus a)$

13. Использование условия включения множеств для упрощения разности.

    $\forall_{abc}(b \subseteq c \to b \setminus (a \cap c) = b \setminus a)$

    $\forall_{abc}(b \subseteq c \to (a \cup b) \setminus c = a \setminus c)$

14. Разность числовых промежутков. $\forall_{abcd}([a,b]\setminus[a,b]=\emptyset)$. На первый взгляд, прием просто дублирует тождество с разностью одинаковых множеств. При переходе от формульного режима просмотра к текстовому выясняем, что второй промежуток - замкнутый (содержит оба конца), а первый - произвольного типа. Несколько спорным является размещение приема в данном разделе, а не в разделе, посвященном числовым промежуткам. Впрочем, это лишь иллюстрирует некоторую хаотичность организации всей базы приемов в целом, являющуюся следствием той интенсивной потоковой проработки последовательности задач, в процессе которой она возникала.

Общая стандартизация утверждений

1. Пустота разности множеств. $\forall_{ab}(a\setminus b=\emptyset\leftrightarrow a\subseteq b)$

2. Включение разности. $\forall_{abc}(a\subseteq b\cup c\leftrightarrow a\setminus b\subseteq c)$. Эквивалентность применяется справа налево. Если включение является условием задачи на описание, причем либо b, либо c содержат неизвестные, то преобразование блокируется. Блокируется оно также при редактировании ответа, если c - неизвестная.

3. Непересечение разности двух множеств с первым из них. $\forall_{ab}(b\subseteq a\leftrightarrow \text{непересек}(b,b\setminus a))$. Эквивалентность применяется справа налево.

4. Равенство разности двух множеств первому из них. $\forall_{ab}(\text{непересек}(a,b)\leftrightarrow a=a\setminus b)$. Эквивалентность применяется справа налево.

5. Включение в разность. $\forall_{abc}(c\subseteq a\setminus b\leftrightarrow c\subseteq a\ \&\ \text{непересек}(b,c))$. Прием аналогичен ранее рассмотренным приемам декомпозиции условий включения в пересечение и включения объединения. В частности, преобразование блокируется, если оно разрушает полученное явное описание для неизвестной c.

6. Принадлежность разности. $\forall_{abc}(a\in b\setminus c\leftrightarrow \neg(a\in c)\ \&\ a\in b)$. Аналогично предыдущему.

7. Непересечение с разностью. $\forall_{abc}(\text{непересек}(c,a\setminus b)\leftrightarrow a\cap c\subseteq b)$. Если преобразуется условие задачи на описание, то либо c известно, либо хотя бы одно из выражений a,b содержит неизвестные. Кроме того, блокируется разрушение явного описания для неизвестной c при редактировании ответа.

Свертка известных утверждений при редактировании ответа

1. Свертка конъюнкции включения и непересечения во включение разности.

   $\forall_{abc}(c\subseteq a\setminus b\leftrightarrow c\subseteq a\ \&\ \text{непересек}(b,c))$

2. Свертка условий включения и непересечения во включение разности. Аналогично предыдущему, но группируются не конъюнктивные члены, а различные условия задачи.

3. Свертка конъюнкции принадлежности и непринадлежности в принадлежность разности. $\forall_{abc}(a\in b\setminus c\leftrightarrow \neg(a\in c)\ \&\ a\in b)$.

4. Свертка условий принадлежности и непринадлежности в приналежность разности. Аналогично предыдущему.

Усмотрение непересечения разности с вычитаемым множеством

$\forall_{ab}(\text{непересек}(b, a \setminus b))$.

Усмотрение непересечения разности и пересечения

$\forall_{abc}(\text{непересек}(a \cap b, c \setminus a))$

Кванторные свертки с участием разности

$\forall_{ab}(a - \text{set} \to \text{непересек}(a, b) \leftrightarrow \exists_c(a = c \setminus b \ \& \ c - \text{set}))$

$\forall_{acd}(\forall_b(\neg(b \in d) \ \& \ b \in c \leftrightarrow b \in a) \leftrightarrow a = c \setminus d)$

Преобразования блокируются при наличии явных указаний (комментарий "кванторнаясвертка" либо цель "попыткапараметризации").

Эквивалентные преобразования условий с неизвестными

1. Группировка условий принадлежности и непринадлежности. $\forall_{abc}(a \in b \setminus c \leftrightarrow \neg(a \in c) \ \& \ a \in b)$. Здесь a - неизвестная; b, c - известны. Эквивалентность применяется справа налево.

2. Группировка условий включения и непересечения. $\forall_{abc}(c \subseteq a \setminus b \leftrightarrow c \subseteq a \ \&$ непересек$(b, c))$. c - неизвестная; a, b известны.

3. Разрешение относительно неизвестной условия включения разности. $\forall_{abc}(b \setminus a \subseteq c \leftrightarrow b \setminus c \subseteq a)$. Выражение c содержит неизвестные, а выражения a, b - нет. Эквивалеэтность применяется справа налево.

4. Попытка представить неизвестную правую часть включения в виде пересечения либо разности. Чтобы обратиться к нормализатору "видразность", выполняющему указанное представление, используется прием: $\forall_{cda}(a = d \to c \subseteq d \leftrightarrow c \subseteq a)$. Заменяющее утверждение обрабатывается нормализаторами "уравнсодержится", "нормлог", и в стандартных случаях прием выдает уже разрешенное относительно неизвестных утверждение.

5. Попытка группировки неизвестных в одном подвыражении условия. Иногда такая группировка позволяет упростить условие относительно неизвестных, например, получить единственное вхождение неизвестной.

   $\forall_{ab}(a \setminus b = \emptyset \leftrightarrow a \subseteq b)$

   $\forall_{abc}(\text{непересек}(c, a \setminus b) \leftrightarrow a \cap c \subseteq b)$

   $\forall_{abc}(a \subseteq b \cup c \leftrightarrow a \setminus b \subseteq c)$

   $\forall_{abc}(b \setminus a \subseteq c \leftrightarrow b \setminus c \subseteq a)$

   В первой теореме a, b содержат неизвестные. Вторая теорема дает два приема: у одного a, c не известны, а b известно, причем эквивалентность применяется слева направо; у другого a, b не известны, а c известно, и эквивалентность применяется справа налево. Аналогичным образом, третья теорема тоже дает два приема. Наконец, в последней теореме b, c не известны; a известно, и эквивалентность применяется слева направо.

Декомпозиция условий задач на доказательство и посылок

Декомпозиция выполняется с помощью следующих эквивалентностей:

$\forall_{bcef}(\text{непересек}(c,(e\cup f)\setminus b)\leftrightarrow \text{непересек}(e\setminus b,c)\ \&\ \text{непересек}(f\setminus b,c))$

$\forall_{acef}(\text{непересек}(c,a\setminus(e\cap f))\leftrightarrow \text{непересек}(a\setminus e,c)\ \&\ \text{непересек}(a\setminus f,c))$

$\forall_{acef}(\text{непересек}(c,a\setminus(e\setminus f))\leftrightarrow \text{непересек}(a\cap f,c)\ \&\ \text{непересек}(a\setminus e,c))$

$\forall_{abcd}(c\subseteq a\cup d\ \&\ b\cap c\subseteq d\leftrightarrow c\subseteq d\cup a\setminus b)$

Попытка преобразования к виду объединения пересечений

Теорема приема имеет вид: $\forall_{abc}(c=a\setminus b\to a\setminus b=c)$. Антецедент присваивает переменной c результат обработки разности нормализаторами "стандобъединение", "группмножество", "свертка". Для срабатывания приема необходимо, чтобы этот результат был короче исходного выражения. Если разность расположена внутри большего теоретико-множественного выражения, то прием блокируется - он будет применен сразу ко всему выражению.

Нормализаторы

Связанные с символом "разность" нормализаторы аналогичны рассмотренным ранее. Это нормализатор "нормразность" (общая стандартизация); "уравнразность" (приведение подобных членов с неизвестными); "видразность" (приведение к виду пересечения либо разности"; "упрощразность" (сокращенная переформулировка).

4.8 Приемы символа "симметричразность"

Для симметрической разности было создано совсем немного приемов, так как она редко встречалась в задачах. Пополнение списка относящихся к ней приемов, по аналогии с уже рассмотренными выше приемами, можно рекомендовать как хорошее упражнение.

Общая стандартизация выражений

1. Одинаковые операнды. $\forall_a(a\bigtriangleup a=\emptyset)$
2. Операнд - пустое множество. $\forall_a(a-\mathrm{set}\to a\bigtriangleup\emptyset=a)$
3. Вложенные операнды. $\forall_{ab}(a\subseteq b\to a\bigtriangleup b=b\setminus a)$
4. Выражение через объединение и разность. $\forall_{ab}(a\bigtriangleup b=a\setminus b\cup b\setminus a)$. Это преобразование, строго говоря, не относится к общей стандартизации. Оно применяется на уровне 5, когда других средств для работы с симметрической разностью не остается. При редактировании ответа задачи на описание прием блокируется. Заметим, что при упрощении выражения симметрическая разность сначала будет устранена данным приемом, а затем, если не произойдут дальнейшие преобразования, - восстановлена приемом нормализатора "упрощобъединение".

Общая стандартизация утверждений

1. Включение симметрических разностей. $\forall_{abc}(a \triangle b \subseteq a \triangle c \leftrightarrow a \cap c \subseteq b \,\&\, b \subseteq a \cup c)$.
2. Непересечение симметрических разностей. $\forall_{abc}(\text{непересек}(a \triangle b, a \triangle c) \leftrightarrow b \cap c \subseteq a \,\&\, a \subseteq b \cup c)$.

Усмотрение симметрической разности

$\forall_{ab}((a \cup b) \setminus (a \cap b) = a \triangle b)$. Прием применяется на уровне 0.

4.9 Приемы символа "непересек"

Ввод специального символа "непересек" для двуместного отношения непересечения множеств оказался более удобным, чем использование записи $a \cap b = \emptyset$. Приемы этого символа во многом аналогичны приемам символа "содержится".

Кванторная расшифровка условия непересечения множеств

$\forall_{ab}(\text{непересек}(a, b) \leftrightarrow \forall_c(c \in a \to \neg(c \in b)))$

Преобразование применяется к условию задачи на доказательство; уровень срабатывания равен 6 (случай корневого вхождения) либо 7. Вводится комментарий "кванторнаясвертка", блокирующий применение обратного преобразования.

Кванторная свертка в условие непересечения множеств

Теорема - та же, что у предшествующего приема, однако эквивалентность применяется справа налево. Уровень срабатывания равен 0. При наличии комментария "кванторнаясвертка" либо при редактировании ответа задачи на описание, имеющей цель "попыткапараметризации", прием блокируется.

Параметрическое описание для условия пересечения множеств

$\forall_{ab}(\neg(\text{непересек}(a, b)) \leftrightarrow \exists_c(c \in a \,\&\, c \in b))$.

Заголовок приема - "параметризация". Применяется он к условию задачи на описание, имеющей цель "пример" либо "параметризация". Вводится вспомогательная задача, в которой данное условие заменяется на утверждения $c \in a, c \in b$, причем переменная c присоединяется к списку неизвестных. Если удается получить ответ, то на него навешивается квантор существования по c. Уровень срабатывания приема равен 6.

Параметрическое описание для условия непересечения множеств

$\forall_{ab}(\text{непересек}(a, b) \leftrightarrow \exists_c(c - \text{set} \,\&\, a = c \setminus b))$.

Аналогично предыдущему, причем дополнительно предполагается, что a - неизвестная, не входящая в b. Уровень срабатывания 5.

Усмотрение пустоты множества из непересечения его с собой

$\forall_a(a = \emptyset \leftrightarrow \text{непересек}(a, a))$

$\forall_{de}(d \subseteq e \ \&\ \text{непересек}(d, e) \to d = \emptyset)$

На второй теореме основаны два приема; у одного проверочным оператором обрабатывается первый антецедент, у другого - второй. Оставшийся антецедент содержится в контексте.

Невключение непересекающихся множеств

$\forall_{AB}(\neg(A = \emptyset) \ \&\ \text{непересек}(A, B) \to \neg(A \subseteq B))$.

Оба антецедента обрабатываются проверочными операторами.

Непересечение вложенных множеств эквивалентно пустоте меньшего из них

$\forall_{de}(d \subseteq e \to d = \emptyset \leftrightarrow \text{непересек}(d, e))$.

Эквивалентность применяется справа налево.

Выдача ответа задачи на описание

Предусмотрены два шаблона ответов с участием символа "непересек". Первый из них должен включать утверждения "множество(x)", "принадлежит(a x)", "непересек(x b)"(возможно, частично), к которым могут добавляться в произвольных количествах утверждения вида "не(содержится(x c))", "не(содержится(d x))". Этот шаблон указывает "нижнюю границу" для неизвестного множества x, в виде элемента a, принадлежащего x, а также "верхнюю границу" в виде множества b, не пересекающегося с x. Если нижняя граница более чем одноэлементна, то она приобретает вид включения. Тогда используется второй шаблон. Он содержит утверждения "множество(x)", "содержится(a x)", "непересек(x b)", а также произвольное число указанных выше дополнительных условий невключения. Оба шаблона реализованы на ГЕНОЛОГе так же, как рассмотренные ранее шаблоы для включения. Теорема приема имеет вид "явное(...)"; заголовок - "ответ(...)". Точка привязки может быть выбрана в любом из трех основных утверждений шаблона, и для каждого выбора существует своя версия приема.

Исключение квантора существования

$\forall_{ab}(\exists_x(x - \text{set} \ \&\ a \subseteq x \ \&\ \text{непересек}(x, b)) \leftrightarrow \text{непересек}(a, b))$

$\forall_a(\exists_x(x - \text{set} \ \&\ \text{непересек}(a, x)))$

Первый прием переформулирует условие существования в терминах непересечения; второй - заменяет квантор существования на константу "истина".

Обращения к проверочным операторам

Чтобы усматривать непересечение двух множеств, создан проверочный оператор "усмнепересек"; чтобы усматривать их пересечение - оператор "усмненепересек". Обращения к ним осуществляются, в частности, из сканирования задачи. Обращение к оператору "усмнепересек" выполняется приемом $\forall_{ab}$(непересек(a, b) $\to$ непересек(a, b)), имеющим заголовок "второйтерм". Уровни срабатывания равны 2 и 4 (последний - для обработки условий задач на описание, содержащих неизвестные). Указатель "коммутативно(фикс(0))" блокирует попытку перестановки операндов a, b при обращении. Эта перестановка будет делаться самим проверочным оператором. Аналогично, для обращения ко второму оператору служит прием $\forall_{ab}(\neg$(непересек$(a, b)) \to \neg$(непересек$(a, b)))$.

Оператор "усмнепересек"

Как и оператор "усмсодержится", данный оператор имеет несколько специальных приемов, понятия которых выходят за рамки раздела "алгебра множеств". По мере обучения системы, число таких приемов может существенно вырасти. Ниже мы ограничимся перечислением лишь общих приемов.

1. Непересечение с пустым множеством.

   $\forall_a$(непересек$(a, \emptyset)$).

   Хотя в теореме приема пустое множество представлено вторым операндом, однако компилятор, учитывая симметричность отношения "непересек", вставляет в программу оператор, перечисляющий оба варианта идентификации - без перестановки операндов и с перестановкой. Поэтому создание альтернативных вариантов приемов, соответствующих такой перестановке, не требуется.

2. Усмотрение из непересечения с надмножеством.

   $\forall_{abc}(a \subseteq b$ & непересек$(b, c) \to$ непересек$(a, c))$. Имеются две версии приема. В каждой из них один антецедент обрабатывается проверочным оператором, а другой содержится в списке посылок.

3. Непересечение с объединением.

   $\forall_{abc}$(непересек(a, c) & непересек$(b, c) \to$ непересек$(c, a \cup b)$).

4. Непересечение с пересечением множеств.

   $\forall_{abc}$(непересек$(a, b) \to$ непересек$(a, b \cap c)$)

5. Декомпозиция условия непересечения с множеством вида $A \cap (B \cup C)$.

   $\forall_{bcef}$(непересек$(b \cap e, c)$ & непересек$(b \cap f, c) \to$ непересек$(c, (e \cup f) \cap b)$)

6. Непересечение с разностью.

   $\forall_{abc}$(непересек$(a, b) \to$ непересек$(a, b \setminus c)$)

   $\forall_{abc}(a \subseteq c \to$ непересек$(a, b \setminus c))$

   $\forall_{abc}(a \cap b \subseteq c \to$ непересек$(a, b \setminus c))$

   В последнем приеме антецедент непосредственно представлен в контексте; в первых двух - обрабатывается проверочным оператором.

7. Декомпозиция условия непересечения с множеством вида $A \setminus (B \setminus C)$.

   $\forall_{acef}$(непересек$(a \cap f, c)$ & непересек$(a \setminus e, c) \to$ непересек$(c, a \setminus (e \setminus f))$)

8. Декомпозиция условия непересечения с множеством $A \setminus (B \cap C)$.

   $\forall_{acef}$(непересек$(a \setminus e, c)$ & непересек$(a \setminus f, c) \to$ непересек$(c, a \setminus (e \cap f))$)

9. Декомпозиция условия непересечения с множеством $(A \cup B) \setminus C)$.

   $\forall_{bcef}$(непересек$(e \setminus b, c)$ & непересек$(f \setminus b, c) \to$ непересек$(c, (e \cup f) \setminus b)$)

10. Непересечение с конечным списком.

    $\forall_{abc}(\neg(a \in c)$ & непересек$(c, \{; b\}) \to$ непересек$(c, \{a; b\})$)

    Указатель "спуск" блокирует альтернативные попытки усмотрения непересечения после того, как было начато рассмотрение элемента a.

11. Прямые произведения.

    $\forall_{ABCD}$(непересек$(A, B) \to$ непересек$(A \times C, B \times D)$)

    $\forall_{ABCD}$(непересек$(C, D) \to$ непересек$(A \times C, B \times D)$)

12. Прообразы.

    $\forall_{adf}$(непересек$(a, d) \to$ непересек(прообраз(f, a), прообраз(f, d)))

13. Непересечение двух множеств, заданных с помощью описателя "класс". Ограничимся рассмотрением простейшего из приведенных здесь приемов:

    $\forall_{fgab}(a = \mathrm{set}_x(f(x))$ & $b = \mathrm{set}_y(g(y))$ & $\neg(g(x)$ & $f(x)) =$ истина $\to$ непересек(a, b)).

    Третий антецедент выделен указателем "идентификатор". Его левая часть обрабатывается нормализатором "нормлог". Если при этом получается константа "истина", то утверждения, определяющие множества, противоречат друг другу.

14. Непересечение с объединением семейства.

    $\forall_{ABP}$(непересек$(A(i), B) \to$ непересек$(\bigcup_{i,P(i)} A(i), B)$)

    Чтобы проверить непересечение множества B с объединением семейства, первый антецедент обращается к проверке непересечения B с произвольным элементом семейства - множеством $A(i)$. Указатель "занесениепосылки(1 значение(x40 x9))" передает вспомогательному проверочному оператору информацию об индексе i - дополнительную посылку $P(i)$.

Проверочный оператор "усмненепересек"

Оператор проверяет непустоту пересечения двух множеств. Его можно было бы назвать и "усмпересек", однако для единообразия выбрана указанная выше версия. Она осталось от серии экспериментов, в которых система автоматически создавала пакетные операторы и выбирала их названия. Этим же экспериментам обязаны своими "странными" названиями такие пакеты, как "видразность"; "видумножение" (последний представляет собой мощный пакет для разложения на множители, который одновременно выполняет сложение дробей) и ряд других.

1. Усмотрение общего элемента. $\forall_{abc}(c \in a \ \& \ c \in b \rightarrow \neg(\text{непересек}(a, b)))$. Второй антецедент в явном виде содержится в контексте; первый - обрабатывается проверочным оператором.

2. Усмотрение из пересечения с подмножеством. $\forall_{abc}(a \subseteq b \ \& \ \neg(\text{непересек}(a, c)) \rightarrow \neg(\text{непересек}(b, c)))$. Две версии приема, в каждой из которых один антецедент содержится в контексте, а другой - обрабатывается проверочным оператором.

3. Пересечение с объединением множеств.

   $\forall_{abc}(\neg(\text{непересек}(a, c)) \rightarrow \neg(\text{непересек}(c, a \cup b)))$.

   Прием просматривает все операнды a, и для каждого из них реализует рекурсивное обращение к проверочному оператору.

4. Пересечение с пересечением множеств.

   $\forall_{abc}(c \subseteq a \ \& \ \neg(\text{непересек}(b, c)) \rightarrow \neg(\text{непересек}(c, a \cup b)))$.

   Оба антецедента обрабатываются проверочными операторами. a последовательно идентифицируется с операндами конъюнкции. Указатель "спуск(1)" означает, что после обнаружения первого же включения альтернативные попытки усмотрения пересечения блокируются.

5. Пересечение с разностью множеств.

   $\forall_{abc}(\text{непересек}(b, c) \ \& \ \neg(\text{непересек}(a, b)) \rightarrow \neg(\text{непересек}(b, a \setminus c)))$.

   $\forall_{abc}(c \subseteq b \ \& \ \neg(c \subseteq a) \rightarrow \neg(\text{непересек}(c, b \setminus a)))$.

6. Пересечение с конечным списком.

   $\forall_{abc}(\neg(\text{непересек}(c, \{;b\})) \rightarrow \neg(\text{непересек}(c, \{a; b\})))$

   $\forall_{abP}(P(a) \rightarrow \neg(\text{непересек}(\{a; b\}, \text{set}_x(P(x)))))$

   Во втором приеме антецедент выделен указателем "очевидно(1)", означающим, что для его обработки применяются проверочные операторы. Отличие от указателя "блокпроверок" состоит в том, что теорема приема не дает возможности непосредственно найти нужный проверочный оператор. Поэтому используется оператор "очевидно", устанавливающий проверочный оператор в зависимости от конкретного проверяемого утверждения.

7. Пересечение двух классов.

   $\forall_{AB}(\neg(\text{set}_x(A(x) \ \& \ B(x)) = \emptyset) \rightarrow \neg(\text{непересек}(\text{set}_x(A(x)), \text{set}_y(B(y)))))$.

   Конъюнкция $A(x)$ & $B(x)$ условий принадлежности двум классам обрабатывается вспомогательной задачей на описание, решаемой с целью "редакция". Роль неизвестных играют переменные связывающей приставки x. Уровень обращения - очень мальнький (равный 2). Непустота результирующего множества проверяется с помощью оператора "усмнепусто".

Нормализатор уравнений "уравннепересек"

Этот нормализатор - последний из используемых в алгебре множеств для явного разрешения утверждения относительно неизвестных. Он аналогичен нормализатору "уравнсодержится".

1. Непересечение множества и его пересечения с другим множеством.

   $\forall_{ab}($непересек$(a,b) \leftrightarrow$ непересек$(a,a\cap b))$.

2. Непересечение множества и результата вычитания из него другого множества.

   $\forall_{ab}(b\subseteq a \leftrightarrow$ непересек$(b,b\setminus a))$.

3. Развертка условия непересечения с объединением.

   $\forall_{abc}($непересек$(c,a\cup b) \leftrightarrow$ непересек(a,c) & непересек$(b,c))$.

   Предполагается, что либо выражение c не содержит неизвестных, либо хотя бы одно из выражений a,b - содержит.

4. Развертка условия непересечения с конечным списком.

   $\forall_{abc}($непересек$(c,\{a;b\}) \leftrightarrow \neg(a\in c)$ & непересек$(c,\{;b\}))$.

   Предполагается, что либо c не содержит неизвестных, либо хотя бы одно из выражений a,b - содержит, либо c имеет своим заголовком один из символов "пересечение", "разность", "прямоепроизведение". Утверждение "$a\in c$" обрабатывается нормализаторами "нормусм", "уравнпринадлежит"; утверждение "непересек$(c,\{;b\})$" - нормализаторами "нормусм", "уравннепересек". Таким образом, в обоих случаях разрешение относительно неизвестных средствами данного пакета доводится до конца. Конъюнкция указанных утверждений дополнительно обрабатывается нормализатором "нормлог".

5. Развертка условия непересечения множества и результата вычитания из конечного списка другого множества.

   $\forall_{abcd}($непересек$(d,\{b;c\}\setminus a) \leftrightarrow (\neg(b\in d)\ \vee\ b\in a)$ & непересек$(d,\{;c\}\setminus a))$.

   Если выражения b,c не содержат неизвестных, то, во-первых, выражение a либо известно, либо имеет своим заголовком один из символов "объединение", "пересечение", "разность", "перечень", "прямоепроизведение", а во-вторых, выражение d либо известно, либо имеет своим заголовком один из символов "пересечение", "разность", "прямоепроизведение". Условия принадлежности и непересечения в заменяющем терме обрабатываются соответствующими нормализаторами уравнений.

6. Развертка условия непересечения множества и пересечения конечного списка с другим множеством.

   $\forall_{abcd}($непересек$(d,a\cap\{b;c\}) \leftrightarrow (\neg(b\in a)\ \vee\ \neg(b\in d))$ & непересек$(d,a\cap\{;c\}))$

   Аналогично предыдущему.

7. Развертка условия непересечения прямых произведений.

   $\forall_{abcd}($непересек$(a\times b,c\times d) \leftrightarrow$ непересек$(a,c)\ \vee\ $ непересек$(b,d))$

   Условия непересечения в правой части обрабатываются нормализатором "уравннепересек".

8. Попытка группировки неизвестных в одной части отношения.

   $\forall_{abc}($непересек$(c,a\cap b) \leftrightarrow$ непересек$(a,b\cap c))$

Эквивалентность применяется справа налево. Выражения a, b содержит неизвестные, причем b идентифицируется со всеми неизвестными операндами пересечения.

$\forall_{abc}(\text{непересек}(c, a \setminus b) \leftrightarrow a \cap c \subseteq b)$

Эквивалентность применяется слева направо. Выражения a, c содержат неизвестные. Результат обрабатывается нормализатором "уравнсодержится", причем перед выполнением замены проверяется, что глубина вхождений неизвестных уменьшилась.

$\forall_{ab}(a \cap b = \emptyset \leftrightarrow \text{непересек}(a, b))$

Эквивалентность применяется справа налево. Выражения a, b содержат неизвестные. Заменяющий терм обрабатывается нормализатором "уравнмножество", и замена выполняется лишь при условии, что глубина вхождений неизвестных уменьшилась.

9. Переход от условия непересечения известного множества с разностью к условию включения пересечения множеств.

   $\forall_{abc}(\text{непересек}(c, a \setminus b) \leftrightarrow a \cap c \subseteq b)$

10. Разрешение относительно неизвестной условия непересечения с пересечением множеств.

    $\forall_{abc}(\text{непересек}(c, a \cap b) \leftrightarrow \text{непересек}(a, b \cap c))$

    Эквивалентность применяется справа налево. Выражение c идентифицируется со всеми неизвестными операндами пересечения; выражение a не содержит неизвестных.

11. Попытка представить содержащую неизвестные часть условия непересечения множеств в виде объединения множеств.

    $\forall_{cda}(a = d \rightarrow \text{непересек}(c, d) \leftrightarrow \text{непересек}(c, a))$

    Антецедент выделен указателем "идентификатор". Он присваивает переменной a результат обработки выражения d нормализатором "видобъединение". Выражение d содержит неизвестные, отлично от переменной и не имеет заголовка "объединение". Проверяется, что a либо имеет заголовок "объединение", либо короче выражения d. Заменяющее утверждение обрабатывается нормализатором "уравннепересек". После этого проверяется, что глубина вхождений неизвестных уменьшилась.

12. Попытка представить одну из частей условия непересечения множеств в виде прямого произведения, если другая часть уже имеет вид прямого произведения.

    $\forall_{cdea}(a = e \rightarrow \text{непересек}(e, c \times d) \leftrightarrow \text{непересек}(a, c \times d))$

    Антецедент присваивает переменной a результат обработки выражения e нормализатором "видпрямоепроизведение", причем не требуется, чтобы это выражение имело неизвестные. В остальном прием аналогичен предыдущему.

4.10 Упражнения

Мы рассмотрели простейшие теоретико-множественные операции и отношения, и прежде чем продолжить перечень приемов по алгебре множеств, предлагаем ряд упражнений. Они позволят лучше разобраться в том, как на самом деле работает перечисленное выше многообразие приемов, а также приобрести первые навыки в самостоятельном программировании на ГЕНОЛОГе.

Упражнения по програмированию на ГЕНОЛОГе обычно приводятся в виде списка задач, для которых недостаточно имеющихся в системе приемов. Однако, по мере развития решателя, потребность в таких приемах может возникнуть, они будут занесены в базу приемов, и приводимые здесь упражнения станут утраченными. Чтобы избежать такой ситуации, в интерфейсе системы предусмотрен специальный ослабленный "учебный" режим запуска задач. В этом режиме блокируется срабатывание приемов ГЕНОЛОГа, имеющих указатель "ученик". По мере возможности, таким указателем будут снабжаться все приемы, воздействующие как-либо на процессы решения приводимых в книге учебных задач. Запуск решения задачи в учебном режиме обеспечивается нажатием клавиши "щ" (без трассировки) либо "Щ" (с трассировкой). Если таким образом запускается решение задачи, ранее хранившейся в задачнике, то данные о ней (ответ и время решения) будут изменены. Чтобы восстановить исходный их вид, достаточно повторно запустить решение задачи нажатием клавиши "о".

Во всех упражнениях этой книги, не оговаривая каждый раз особо, запускаем решатель только нажатием клавиши "щ" либо "Щ". Если в решателе обнаруживаются приемы с указателем "ученик", о которых в соответствующих разделах книги ничего не говорилось, то они были созданы уже после написания данных разделов.

1. Найти первую задачу подраздела задачника "Теория множеств - Задачи на преобразование" и просмотреть процесс ее решения по шагам, восстанавливая все применяемые преобразования. В том числе, определить все преобразования нормализаторов общей стандартизации (в общем режиме трассировки "по срабатываниям приемов" они на экран не выводятся).

2. Найти в разделе "Теория множеств - Задачи на описание" задачу номер 15 и просмотреть процесс ее решения по шагам, восстанавливая все применяемые преобразования.

3. Найти в разделе "Теория множеств - Задачи на доказательство" задачу номер 21 и просмотреть процесс ее решения по шагам, восстанавливая все логические переходы.

4. Ввести приемы, позволяющие решать задачи на преобразование выражений алгебры множеств к виду объединения пересечений. Использовать их для приведения к указанному виду выражения $(a \cup e \cup (b \cap c)) \cap (c \cup d \setminus b \cup f)$.

5. Применить решатель для упрощения выражения, полученного в предыдущем примере после "раскрывания скобок". Убедиться в том, что результат оказывается сложнее исходного выражения предыдущего примера, и добавить приемы, необходимые для восстановления этого выражения.

6. Добавить приемы, необходимые для решения задачи на нахождение всех таких множеств A, B, что выполнено условие $\exists_{xy}(A \setminus x \subseteq B \cap y \ \& \ B \setminus x \subseteq A \cap y)$.

7. Добавить приемы, необходимые для решения задачи на нахождение всех таких множеств A, B, C, что выполнено условие $\exists_x(\forall_y(C \subseteq y \to A \cup x \subseteq B \cap y))$.

8. Добавить приемы, необходимые для решения задачи на упрощение выражения $(C \cup D) \setminus A$ при выполнении посылок $A \subseteq B$, $\forall_x(A \subseteq x \;\&\; x \subseteq B \to C \setminus x = D \setminus x)$.

9. Добавить приемы, необходимые для решения задачи на нахождение примера таких значений x, y, z, что $x \subseteq y, y \subseteq z, \neg(y \subseteq x), \neg(z \subseteq y)$.

10. Найти еще какую-нибудь задачу на рассмотренные понятия алгебры множеств, не решаемую системой, и добавить необходимые для ее решения приемы.

4.10.1 Указания

1. Запускаем решение задачи с трассировкой "по срабатываниям приемов" (клавиша "p"). Первое преобразование заключается в устранении вложенных пересечений. Следующий переход - замена выражения $(a \cup b \setminus d) \cap (a \setminus (b \cup c)) \cap (d \setminus (a \cap b))$ на выражение $(a \cap d) \setminus (b \cup c)$. Прием, выполняющий данную замену, основан на теореме $\forall_{abc}(c \cap (b \setminus a) = (b \cap c) \setminus a)$. Для просмотра этой теоремы используем клавишу "б"; для возвращения к текущему кадру трассировки - клавишу "End". Сразу понятно, что основная часть содержательных преобразований была выполнена нормализаторами, к которым обращался прием. Это, безусловно, ускорило вычисления, но создало трудности в пошаговом просмотре решения.

   Если бы прием пользовался какими-либо крупными нормализаторами, формат которых содержит элемент "контрольнормализации", то можно было бы обратиться из текущего кадра трассировки к детальному просмотру цепочки их преобразований. Однако, в нашем случае использовались только нормализаторы общей стандартизации. На момент просмотра текущего кадра обращения к ним уже состоялись. Поэтому нужно повторить трассировку решения задачи с самого начала и постараться войти в пошаговый анализ работы нормализаторов рассматриваемого приема через отладчик ЛОСа. Выполняем следующие действия. Возвращаемся к просмотру задачи (Esc) и входим в оглавление приемов для выбора точки прерывания (клавиша "г"). Оказываемся в оглавлении базы приемов, у которого выделен пункт "Пересечение с разностью", содержащий нужный нам прием. Нажимаем "курсор вправо" и входим в просмотр данного приема. Для выбора точки прерывания перед его реализацией и запуска решения нажимаем "Ctr-Enter". Почти сразу появляется кадр отладчика ЛОСа, соответствующий начальным операторам приема. Из программы приема видно, что обращения к нормализаторам расположены внутри оператора "заменавхождения(...)". Поэтому выбираем точку прерывания перед реализацией данного оператора: нажимаем "курсор вниз"; с помощью клавиши "курсор вправо" выделяем синим цветом оператор "равно(x10 набор(x8))", предшествующий оператору "заменавхождения", и нажимаем "Enter". Отладчик переходит в кадр, непосредственно предшествующий обращениям к нормализаторам. Убеждаемся в том, что преобразуется нужный нам терм, выведя его на экран, например, путем просмотра значения переменной x2 - текущего вхождения в задачу. Из программы видно, что прежде всего будет применяться нормализатор "нормпересечение". Поэтому устанавливаем прерывание при обращении к нему: нажимаем клавищу "л"; вводим текстовым редактором символ "нормпересечение" и нажимаем "Enter". Появляется стартовый кадр работы нормализатора.

Чтобы просмотреть цепочку изменений преобразуемого выражения, нажимаем клавиши "7" (появляется буква "з" и курсор текстового редактора), "1"(номер той переменной, изменения которой нас интересуют), и "Enter". Таким образом создана установка на просмотр изменений переменной x1, и каждое следующее нажатие клавиши "Enter" будет приводить к кадру, указывающему очередное изменение. Нажимаем эту клавишу. На экране появляется текст "Условие $a \cap (a \cup b \setminus d) \cap (d \setminus (a \cap b))$ заменяется на $(d \setminus (a \cap b)) \cap a$ (Безусловное поглощение)". Клавиша "б" позволяет посмотреть описание примененного приема. Это преобразование уже действительно одношаговое: второй член пересечения поглощается первым. Снова нажимаем "Enter". Появляется очередной кадр: "Условие $(d \setminus (a \cap b)) \cap a$ заменяется на $(a \cap d) \setminus b$ (Пересечение с разностью)". Хотя это преобразование и несложное, оно уже не одношаговое.

Если бы мы захотели разложить его на составные шаги, то пришлось бы еще раз повторить трассировку, и на предыдущем кадре цепочки преобразований нормализатора установить прерывание перед входом в данный прием. Впрочем, есть и другие способы выйти на нужные точки преобразований с помощью отладчика ЛОСа. Можно вставить в программу представляющего интерес приема оператор "трассировка(стоп 0)" чтобы отслеживать все моменты его применения; если нужно отслеживать лишь часть таких моментов, то вставить оператор "или(A трассировка(стоп 0))", где A - оператор, ложный для представляющих интерес случаев, и т.п.

Продолжая начатую выше трассировку, нажимаем "Enter" и попадаем в кадр "Ответ: $(a \cap d) \setminus b$". Чтобы просмотреть теперь работу другого нормализатора, нажимаем "ф" и попадаем снова в отладчик ЛОСа. С помощью нажатия клавиши "PageUp" возвращаемся в просмотр программы приема сканирования задачи. Из нее усматривается, что далее будет применяться нормализатор "нормразность". Как и выше, входим в его стартовый кадр, устанавливаем трассировку по шагам изменения переменной x1 и прослеживаем остающиеся преобразования. В конечном счете, оказываемся в кадре трассировки задачи "по срабатываниям приемов", с которого и начали описанную цепочку действий.

Заметим, что обычно не возникает потребность в прослеживании вычислений, выполняемых нормализаторами общей стандартизации. Лишь в редких предметных областях, типа алгебры множеств, такие нормализаторы могут выполнять сколь-нибудь неочевидные действия. Поэтому специальный интерфейс слежения за их действиями и не был создан.

2. Задача имеет условия $a \cup x \setminus y = b \cap x$, $x \setminus a = x \cap y$ и неизвестные x, y. Запускаем процесс ее решения нажатием клавиши "р". Сразу же срабатывает прием, предпринимающий попытку разрешить оба уравнения сначала относительно y. Это - общелогический прием, реализованный на ЛОСе. Сопровождающий срабатывание текст "решение задачи на описание с несколькими неизвестными путем выражения одной из неизвестных через остальные" попросту копирует название концевого пункта оглавления программ, соответствующего ближайшей внешней контрольной точке "прием(...)" текущего оператора программы. Изменяя название пункта оглавления, будем изменять и сопровождающий текст.

   Следующее нажатие "Enter" приводит к кадру, заменяющему первое уравнение системы на конъюнкцию утверждений $a \subseteq b$, $a \subseteq x$, $x \setminus b \subseteq y$, непересек$(y, (b \cap x) \setminus a)$. Они явно разрешают первое уравнение относительно y. Примененный прием заменил равенство на два включения, а каждое включение обработал нормализатором "уравнсодержится". В отличие от предыдущего примера, нет необходимости повторно запускать решение задачи, чтобы посмотреть шаги работы данных нормализаторов. Прокручивая текущий кадр вниз, находим окна, соответствующие обращениям к нормализаторам. Выбираем, например, первое из них и располагаем так, чтобы верхняя горизонтальная линия окна находилась в верхней части экрана, причем над ней не оставалась прорисованной часть предыдущего окна. Для этого удобно применять клавиши "Ctr-курсор вверх" и "Ctr-курсор вниз". В окне содержится текст "Применить нормализатор "уравнсодержится" к выражению: $a{\cup}x{\setminus}y \subseteq b{\cap}x$". Для повторения действий нормализатора нажимаем "Enter". Первый шаг состоит в развертке условия включения объединения. Оно преобразуется в конъюнкцию $a \subseteq b \;\&\; a \subseteq x \;\&\; x{\setminus}b \subseteq y$. Впрочем, здесь снова имелись обращения к нормализаторам. Окна этих обращений размещены снизу, и можно просмотреть цепочки их преобразований так же, как для текущего нормализатора. Глубина вложенности таких просмотров не ограничена. Если нужно оборвать просмотр на текущем уровне и вернуться к предыдущему уровню, нажимаем "End". Иначе возвращение произойдет автоматически, по нажатии "Enter" при получении ответа на текущем уровне.

Возвращаемся к прерванному просмотру основной траектории преобразований задачи. Следующее нажатие "Enter" дает разрешение относительно y второго уравнения. После цепочки несложных преобразований далее возникает кадр с сопровождающим текстом "Учет неизвестных, которые временно рассматривались как известные". Прокруткой выводим на экран расположенный над ним кадр, описывающий текущее состояние задачи: $x \setminus a \cup x \setminus b \subseteq y, \text{непересек}(y, b \cap x), a \subseteq x, a \subseteq b$. Видно, что относительно y имеем явное описание. Возникший кадр вызван срабатыванием приема усмотрения ответа. Этот прием обратился к процедуре "редакторответа", а она обнаружила в комментариях информацию о том, что переменная x представляет собой "отложенную" неизвестную. Поэтому была введена вспомогательная задача разрешения относительно x двух не зависящих от y утверждений $a \subseteq x, a \subseteq b$, к которым добавлено условие существования значений y, удовлетворяющих найденному явному описанию. Последнее сохраняется отдельно и будет впоследствии добавлено к ответу данной вспомогательной задачи.

Нажимаем "Enter" и входим в решение задачи с неизвестной x. Сразу срабатывает прием, устраняющий квантор существования. Он заменяется на утверждение $\text{непересек}(x{\setminus}a{\cup}x{\setminus}b, b{\cap}x)$. Далее идет цепочка несложных преобразований, приводящих к следующему ответу для x:

$a \subseteq x, a \subseteq b, \text{неперсесек}(x, b \setminus a)$.

Наконец, возникает кадр обращения к вспомогательной задаче на редактирование объединенного ответа. В этом ответе имеется включение $a \subseteq b$, отсутствовавшее на момент разрешения уравнений относительно y. Теперь оно дает возможность упростить условие $x{\setminus}a{\cup}x{\setminus}b \subseteq y$, которое заменяется на $x{\setminus}a \subseteq y$. Затем выдается окончательный ответ.

3. Задача на доказательство имеет своими посылками утверждения $a \cap b = \emptyset, c \subseteq b, c \subseteq d, b \subseteq e, d \cap e \cap f \subseteq a$. Условием ее является утверждение $c \cap f = \emptyset$. Запускаем решение нажатием "р". Прежде всего, посылка $a \cap b = \emptyset$ заменяется на "непересек(a, b)", а условие - на "непересек(c, f)". Следующий кадр - расшиф-

ровка условия непересечения: оно заменяется на $\forall_g(g \in c \to \neg(g \in f))$. Еще одно нажатие "Enter" - и срабатывает прием, усматривающий ложность принадлежности $g \in f$. В результате условие задачи заменяется на $\forall_g(g \in c \to \neg$ложь$)$. Прием использовал проверочный оператор "усмнепринадлежит". Так как после действий этого оператора задача, по существу, оказывается решенной, рассмотрим их подробнее. Прокручиваем изображение до тех пор, пока не появится окно "Проверить утверждение $\neg(g \in f)$. Это и есть обращение к проверочному оператору. Добиваемся расположения верхней горизонтальной линии окна в верхней части экрана и нажимаем "Enter". В отличие от работы нормализатора, представляющей собой цепочку шагов, работа проверочного оператора сводится к единственному шагу - срабатыванию приема, усматривающего истинность проверяемого утверждения. Поэтому текущий кадр уже дает ответ на вопрос о том, как была усмотрена истинность утверждения $\neg(g \in f)$. Текст "использование включения множеств" во втором сверху окне является ссылкой на примененный прием. Нажимаем клавишу "б" и находим теорему приема: $\forall_{abcd}(a \cap b \subseteq c \,\&\, d \in a \,\&\, \neg(d \in c) \to \neg(d \in b))$. Для возвращения нажимаем "End". Чтобы понять, как эта теорема была применена, нужно вывести на экран список посылок проверочного оператора. Выделяем правой кнопкой мыши верхнее окно, содержащее текст "Повторное применение процедуры", и нажимаем клавишу "п". Появляется искомый список, предшествующий проверяемому утверждению. Из него видно, что было использовано включение $d \cap e \cap f \subseteq a$. Окна обращений к вспомогательным проверочным операторам показывают, что использовались также утверждения $g \in d \cap e, \neg(g \in a)$. Можно войти в просмотр проверки этих утверждений, и т.д. Таким образом, трассировка проверочных операторов выполняется в обратном порядке - от заключения к посылкам. Рекомендуется довести ее в данном примере до конца самостоятельно.

4. Решатель имеет нормализатор "стандобъединение", выполняющий требуемые преобразования. Однако, пока нет целевой установки задач на преобразование выражений к виду объединения пересечений, и нет приема, который при наличии такой установки обращался бы к нормализатору "стандобъединение". Мы должны ввести то и другое. Прежде всего, выбираем логический символ для цели задачи. Например, можно взять в качестве такого символа название нормализатора "стандобъединение". Предварительно проверяем, что он еще не использован в целевых установках. Для этого входим в редактор ЛОСа через символ "стандобъединение" и нажимаем F3. Попадаем в окно справочной информации по символу, которое оказывается пустым. Сохраняем в нем краткий текст об использовании символа в качестве цели задач на преобразование.

   Теперь нужно создать прием, обращающийся к оператору "стандобъединение" в задачах на преобразование, имеющих одноименную цель. Сразу же возникает вопрос о символе, при появлении которого в задаче должна выполняться попытка применения приема. Можно было бы отказаться от фиксации этого символа вообще - в произвольной задаче на преобразование выполнять попытку преобразований, если имеется цель "стандобъединение". Однако, утяжелять без особой необходимости группу общих приемов нежелательно - лучше создать несколько приемов, связанных с различными символами частной предметной области, и инициировать попытки применения приема только для задач этой области. В нашем случае придется создать приемы, связанные с теоретико-множественными операциями "пересечение", "объединение" и "разность". Они устроены почти одинаково, так что можно ввести какой-либо один, а прочие получать из его копий небольшими коррекциями. Такая практика при программировании на ГЕНОЛОГе применяется сравнительно часто.

Входим в оглавление базы приемов и выбираем подходящий раздел - пусть, например, это будет подраздел "объединение" раздела "Теория множеств". Создаем новый концевой пункт - например, "Обращение к нормализатору СТАНДОБЪЕДИНЕНИЕ". Входим в этот пункт и начинаем ввод теоремы приема. Нажимаем "Ctr-ф", после чего возникает курсор формульного редактора. Вводим теорему приема (например, для символа "пересечение"): $\forall_{abc}(c = a \cap b \to a \cap b = c)$. Антецедент теоремы означает присвоение переменной c результата применения нормализатора "стандобъединение" к выражению $a \cap b$. Обращение к этому нормализатору будет введено ниже. В приниципе, можно было бы не использовать вспомогательной переменной c и ввести теорему $\forall_{ab}(a \cap b = a \cap b)$, снабдив ее тем же самым обращением к нормализатору. Однако, такой стиль программирования на ГЕНОЛОГе нежелателен - вспомогательные переменные для результатов промежуточных преобразований упрощают ссылки на них из фильтров приема.

По завершении ввода формулы нажимаем "Enter". Под теоремой прорисовывается горизонтальная черта. Теперь нужно ввести заголовок приема. Так как речь идет о приеме замены "слева направо", то нужный заголовок - "второйтерм". Чтобы ввести его, нажимаем снова "Enter", и далее пользуемся текстовым редактором.

Еще одно нажатие "Enter", и появляется третье окно. В нем нужно ввести фильтры приема. Обязательно нужно указать уровень срабатывания приема. Например, вводим текст "уровень(2)". Далее указываем целевой контекст: "тип(преобразовать)", "цель(стандобъединение)", "корень". Последний фильтр разумно ввести, чтобы прием применялся лишь ко всему терму, а не к его многочисленным подтермам. Нажимаем "Enter" и переходим к набору содержимого четвертого окна.

Выделяем антецедент теоремы указателем "идентификатор(1)" и нажимаем "Enter". Появляется голубая горизонтальная черта. Она означает, что введенный прием пока не сохранен. Его нужно немедленно сохранить, нажав F4. Иначе, при случайном выходе из текущего кадра (например, в оглавление базы приемов), вся проделанная работа будет утеряна. После того, как прием сохранен, нижняя черта становится красной. Это значит, что прием пока не откомпилирован. Перед компиляцией нужно еще ввести обращение к нормализатору. Выделяем в антецеденте подтерм $a \cap b$ - мышью либо клавишами курсора, либо и тем, и другим. Затем нажимаем "Enter". Под красной чертой появляется курсор текстового редактора. Вводим название нормализатора "стандобъединение" и нажимаем "Enter". Чтобы отменить выделение подтерма, нажимаем "пробел". Теперь прием можно откомпилировать, нажав клавишу F3. Одновременно будет сохранено изменение приема - добавление к нему нормализатора. Если здесь компилировать не через F3, а через F5, то изменение будет утеряно. Рекомендуется ознакомиться с результатом компиляции - нажать "Home" для перехода в просмотр программы ЛОСа, а затем вернуться в ГЕНОЛОГ через "End".

Теперь переходим к проверке работы созданного приема. Входим в задачник и вводим указанную в упражнении задачу. Целевую установку ее создаем с помощью текстового редактора: нажимаем "Ctr-ц" и далее вводим текст "преобразовать стандобъединение одз". Условие $(a\cup e\cup(b\cap c))\cap(c\cup d\setminus b\cup f)$ вводим обычным образом. Применяя трассировку "по приемам", на первом же шаге обнаруживаем срабатывание созданного нами приема. Рекомендуется провести трассировку по шагам работы нормализатора "стандобъединение", войдя в нее через отладчик ЛОСа. Для этого придется заново запустить решение задачи.

Если бы мы ввели в целевую установку задачи также символ "упростить", то обнаружили бы зацикливание: после каждого раскрывания скобок снова выполняется обратная "свертка" выражения. Рекомендуется немного усовершенствовать прием, чтобы в таких ситуациях (впрочем, уже по постановке задачи противоречивых) зацикливание не возникало.

Вернемся к трассировке решения нашей задачи. После следующего нажатия "Enter" обнаруживаем, что требуемый вид выражения нарушен: произошло обращение к процедуре завершающего редактирования ответа, которая решила вынести часть переменных за скобки. Это противоречит цели "стандобъединение", и нужно заблокировать обращение к указанной процедуре. Чтобы найти ее программу, не выходя из трассировки нажимаем "б". На экране появляется концевой пункт оглавления программ "Завершающее редактирование ответа". Он будет сохранен в оглавлении программ как "текущий пункт". Чтобы войти в редактирование программы этого пункта, возвращаемся к кадру трассировки (клавиша "End"); обрываем трассировку ("Esc") и выходим на главное меню ("End"). Далее входим в оглавление программ ("л") и снова оказываемся на пункте "Завершающее редактирование ответа". Нажимаем "курсор вправо", чтобы выйти в просмотр программы. Сбрасываем режим выделения операторов нажатием клавиши "о", и для редактирования программы нажимаем "р". В начале программы уже имеется цепочка операторов вида "не(цель(x1 ...))". Добавляем к ним оператор "не(цель(x1 стандобъединение))". При повторной трассировке решения задачи убеждаемся, что требуемый вид ответа не нарушен.

5. Чтобы создать задачу на упрощение полученного выше выражения, выполняем следующие операции: выделяем левой кнопкой мыши ответ предыдущей задачи; высвобождаем на экране место для новой задачи и нажимаем "з" (появляется верхняя линия новой задачи); вводим цель "Упростить в о.д.з."; выделяем новую задачу правой кнопкой мыши; нажимаем "Insert" для копирования выделенного ответа в качестве условия новой задачи. Запускаем решение задачи нажатием "о" и получаем ответ $a\cap(c\cup f)\cup c\cap(b\cup e)\cup e\cap f\cup(d\cap(a\cup e))\setminus b$. Очевидно, он сложнее исходного выражения. Нужно подобрать одно или несколько простых тождеств, которые могли бы продолжить упрощение, и создать их приемы. Для первых трех членов указанного выше объединения можно усмотреть возможность следующей перегруппировки: $(a\cup e)\cap(c\cup f)\cup b\cap c$. Это подсказывает такое тождество: $\forall_{abcde}((a\cap(b\cup c))\cup(b\cap(d\cup e))\cup c\cap d=((a\cup d)\cap(b\cup c))\cup c\cap d)$. Рекомендуем читателю создать соответствующий прием и посмотреть, к чему приводит его срабатывание.

6. Вводим задачу на описание, имеющую неизвестные A, B и единственное условие $\exists_{xy}(A\setminus x\subseteq B\cap y\ \&\ B\setminus x\subseteq A\cap y)$. Начинаем трассировку ее решения.

В некоторый момент срабатывает прием, предпринимающий попытку явного разрешения подкванторного утверждения относительно x, y. Продолжаем трассировку этой попытки. Сначала список неизвестных сужается до единственной неизвестной y; после разрешения относительно y - сопровождающие утверждения разрешаются относительно x. В заключение предпринимается редактирование объединенного ответа $A \setminus x \cup B \setminus x \subseteq y, A \setminus B \cup B \setminus A \subseteq x$. Заметим, что обе неизвестные x, y здесь - несущественные, так как имеется цель (параметры xy). Поэтому они могли бы быть устранены, однако приемов для этого нет - решатель ограничивается переходом к симметрической разности $A \triangle B$ и выдает ответ $A \triangle B \subseteq x, x - \text{set}, y - \text{set}, (A \cup B) \setminus x \subseteq y$. Далее происходит возвращение к исходному условию существования, которое теперь приобретает вид $\exists_{xy}(A \triangle B \subseteq x, x - \text{set}, y - \text{set}, (A \cup B) \setminus x \subseteq y)$. Здесь опять появляется очевидная возможность исключить сначала y, а затем x. Однако, приема для этого нет, и в некоторый момент решатель переходит к неявной зависимости от y, заменяя включение $(A \cup B) \setminus x \subseteq y$ на $A \cup B \subseteq x \cup y$. Преобразования такого рода продолжаются; в конце концов существование требуемых значений x, y становится неочевидным, и решатель выдает "отказ".

Приведенная траектория позволяла решателю в двух точках добиться успеха. В первый раз - при редактировании ответа вспомогательной задачи мог сработать прием, исключающий несущественную неизвестную X, для которой имеются лишь условия вида $P \subseteq X$, $X - \text{set}$. Второй раз - при рассмотрении квантора существования можно было из тех же соображений отбросить связанные переменные. Хотя для данной задачи достаточно было бы лишь одного из таких двух приемов, однако каждый из них мог бы быть полезен в других ситуациях независимо от другого. Первый прием основан на теореме $\forall_a(a - \text{set} \rightarrow \exists_x(a \subseteq x \ \& \ x - \text{set}))$. Заголовок приема - "связка". Фильтры - "уровень(0)", "тип(описать)", "не(входит(x23 x1))". Второй прием основан на теореме $\forall_{AB}(\exists_{xy}(x - \text{set} \ \& \ A(y) \subseteq x \ \& \ B(y)) \leftrightarrow \exists_y(B(y)))$. Он имеет заголовок "второйтерм"; фильтр "уровень(0)" и указатели "отображение(x26 x27)", "кортежпеременных(x24)". Последний указатель допускает пустоту списка y, так что прием будет работать и для единственной переменной x. Рекомендуем самостоятельно создать оба приема и проверить, что они позволяют решить задачу.

7. Создаем задачу на описание, имеющую условие $\exists_x(\forall_y(C \subseteq y \rightarrow A \cup x \subseteq B \cap y))$ и неизвестные A, B, C. Начинаем трассировку решения этой задачи. После ряда общелогических преобразований в условии появляется подутверждение $\forall_y(y - \text{set} \rightarrow \neg(C \subseteq y))$. Оно сохраняется достаточно долго, чтобы понять, что решатель не имеет приема, усматривающего его ложность. Отсутствие такого приема доводит дело даже до разбора случаев. Поэтому первый шаг проработки задачи ясен - нужно ввести прием, основанный на теореме $\forall_A(\neg(\forall_x(x - \text{set} \rightarrow \neg(A \subseteq x))))$. Прием имеет заголовок "второйтерм"; уровень срабатывания его выбираем равным 0. Введя прием и откомпилировав его, повторяем трассировку задачи. После цепочки преобразований возникают три условия: $A \subseteq B$, $\exists_x(\forall_y(y - \text{set} \ \& \ C \subseteq y \rightarrow x \subseteq y) \ \& \ x - \text{set} \ \& \ x \subseteq B)$, $\forall_y(y - \text{set} \ \& \ C \subseteq y \rightarrow A \subseteq y)$. Здесь срабатывает прием, обращающийся к вспомогательной задаче для разрешения относительно x утверждения под квантором существования. Она имеет условия $\forall_y(y - \text{set} \ \& \ C \subseteq y \rightarrow x \subseteq y)$ и $x \subseteq B$. Попытка разрешить относительно y утверждения под квантором общности ни-

чего не дает - после ряда преобразований этот квантор возникает опять. На некотором уровне всплывает попытка решать задачу накоплением необходимых условий, пока они не сложатся в достаточные. Она тоже безуспешна. С другой стороны, рассматриваемая кванторная импликация, очевидно, эквивалентна условию включения множества x в множество C. Эта эквивалентность представляется настолько простой, что для нее стоит ввести отдельный прием. Он имеет теорему $\forall_{AB}(A-\text{set}\ \&\ B-\text{set} \rightarrow \forall_x(x-\text{set}\ \&\ A \subseteq x \rightarrow B \subseteq x) \leftrightarrow B \subseteq A)$, заголовок приема - "воройтерм". Антецеденты обрабатываются проверочными операторами; уровень срабатывания выбираем равным 1. После ввода и компиляции приема еще раз повторяем трассировку задачи. Теперь уже она доводится до ответа: $A-\text{set}, B-\text{set}, A \subseteq B \cap C, C-\text{set}$.

Рекомендуем далее отключить последний прием и попробовать добавить приемы, с помощью которых решатель смог бы реализовать попытку накопления необходимых условий, пока они не сложатся в достаточные.

8. Вводим задачу на преобразование, имеющую посылки $A \subseteq B$, $\forall_x(A \subseteq x\ \&\ x \subseteq B \rightarrow C \setminus x = D \setminus x)$ и условие $(C \cup D) \setminus A)$. Очевидно, что для ее решения нужно сначала использовать кванторную импликацию и вывести равенство $C \setminus A = D \setminus A$, а затем усмотреть возможность применить данное равенство к преобразуемому выражению. Пока решатель не делает ни того, ни другого. Как мы знаем, имеется общелогический прием, предпринимающий попытку использовать кванторную импликацию для вывода следствий. Однако, для его срабатывания требуется, чтобы выводимое утверждение не содержало новых выражений. Поэтому сначала обеспечим вывод вспомогательных посылок "актив($C \setminus A$)", "актив($D \setminus A$)", указывающих, что данные выражения представляют интерес. Теорема приема имеет вид $\forall_{AB}$(актив$(A \setminus B)$); заголовок приема - "вывод". Чтобы активизировать прием при обнаружении в условии задачи выражения $(A \cup C) \setminus B$, вводим указатель "контрольвывода(разность(объединение(x26 x28)x27))". Указываем в фильтрах контекст срабатывания: "уровень(2)", "тип(преобразовать)", "цель(упростить)". Далее указываем, что задача имеет кванторную посылку, консеквент которой содержит разность, пересекающуюся по переменным с преобразуемым выражением: "контекст(посылка(x3)заголовок(x3 длялюбого)последнийсимвол(x3 равно) подчинено(x4 последнийоперанд(x3))символ(x4 разность)пересекаются(параметры(x4)параметры(теквхожд)))".

   После компиляции приема убеждаемся в выводе необходимых утверждений. Заметим, что эти утверждения на экране не прорисовываются. Чтобы их увидеть, нужно перейти к показу полного списка посылок нажатием клавиши "ы". Впрочем, косвенной информацией о их появлении служит вывод следствия $C \setminus A = D \setminus A$.

   Следующий шаг - использование выведенного равенства - можно обеспечивать различными средствами. Ограничимся простейшим - создадим прием с теоремой $\forall_{ABCDE}(A \setminus B = C\ \&\ E = C \cup D \setminus B \rightarrow (A \cup D) \setminus B = E)$. Он усматривает посылку $A \setminus B = C$ и предпринимает попытку применить нормализаторы общей стандартизации к выражению, полученному из $(A \cup D) \setminus B$ заменой "неявного" подвыражения $A \setminus B$ на C. Если результат E получился короче исходного выражения, то реализуется замена. Заголовок приема - "воройтерм"; фильтры - "уровень(3)", "короче(x30 фикс(0 1))"; указатель - "идентификатор(2)". Выражения $D\backslash B$ и $C\cup(D\backslash B)$ обрабатываются соответствующими нормализаторами общей стандартизации.

Конечно, предложенный "учебный" вариант проработки предложенной задачи не претендует на особую эффективность и общность. Можно предложить читателю развить в решателе общий аппарат для решения задач указанного типа - на упрощение выражений алгебры множеств при наличии в посылках дополнительных соотношений. Здесь был бы полезен специальный нормализатор, выделяющий в преобразуемом выражении вхождения заданного подвыражения, извлеченного из внешнего контекста.

9. Вводим задачу на описание, имеющую неизвестные x, y, z и условия $x \subseteq y$, $y \subseteq z$, $\neg(y \subseteq x)$, $\neg(z \subseteq y)$. Цели ее определяем с помощью пункта "Найти пример значений неизвестных" оглавления целевых установок. При трассировке решения обнаруживается, что происходит последовательный ввод параметрических описаний, преобразующих задачу к новым неизвестным. Например, условие $x \subseteq y$ и неизвестная y исключаются с помощью параметрического описания, представляющего y как $x \cup a$; аналогичным образом исключаются условие $x \subseteq z$ и неизвестная z, и т.д. По ходу дела предпринимается разбор случаев. Доведя трассировку до получения первого отказа на один из подслучаев, обнаруживаем задачу с условиями $\neg(f = c)$, $\neg(f \in x)$, $\neg(f \in d)$, $\neg(c \in x)$, $g \subseteq g \cup x \cup \{c, f\}$. Видно, что для условий непринадлежности параметрические описания введены не были, и это подсказывает ввод соответствующего приема. Его теорема имеет вид $\forall_{bc}(c - \text{set} \to \neg(b \in c) \leftrightarrow \exists_a(a - \text{set} \ \& \ c = a \setminus \{b\}))$; заголовок приема - "параметризация". Фильтры приема вводим по аналогии, например, с приемом, дающим параметрическое описание условия принадлежности: "уровень(5)", "условие", "тип(описать)", "или(цель(пример) цель(параметризация)) неизвестная(x3) не(входит(x3 x2))". Антецедент выделяем указателем "блокпроверок(1)".

   После создания указанного приема повторяем трассировку до получения первого отказа. Оказывается, что на этот раз не решена задача подбора примера таких c, g, что $\neg(c = g)$. Вводим прием с теоремой $\forall_{ab}(a = 0 \ \& \ b = 1 \ \& \ \to \neg(a = b))$ и заголовком "подборзначений". Фильтры приема определяем так: "уровень(1) условие тип(описать) цель(пример) неизвестная(x1) неизвестная(x2) не(контекст(новоеусловие(x3) или(входит(x1 x3) входит(x2 x3))))". Указатель "подборзначений(1 2)" показывает, что оба антецедента используются для замещения в задаче условия $\neg(a = b)$. Указатель "новый" стоит ввести для того, чтобы отслеживать момент исключения прочих условий задачи, содержащих переменные a, b.

   После создания указанного приема решатель доводит задачу до ответа: $z = \{0, 1\}, y = \{1\}, x = \emptyset$.

4.11 Приемы для работы с конечными списками

Продолжаем рассмотрение приемов для понятий, относящихся к алгебре множеств. В этом разделе речь пойдет о символе "перечень", позволяющем определять конечное множество по заданному упорядоченному набору объектов. Во внутреннем представлении решателя используется запись "перечень(A)", где A - выражение, определяющее набор. Обычно оно имеет своим заголовком символ "набор", и тогда конечное множество с элементами $A_1, \dots A_n$ определяется выражением "перечень(набор($A_1 \dots A_n$))". Формульный редактор прорисовывает это выражение в виде $\{A_1, \dots, A_n\}$. Фактически запятые относятся к представлению набора $(A_1 \dots A_n)$, внешние скобки которого вообще как правило опускаются, а фигурные скобки относятся к символу "перечень". Если нужно прорисовать выражение "перечень(A)", где A не имеет заголовка "набор", то формульный редактор выдает на экран $\{; A\}$. Выражение "перечень(префикс(A B))" прорисовывается в виде $\{A; B\}$; выражение "перечень(конкатенация(набор(A B)C))" - в виде $\{A, B; C\}$.

Общая стандартизация выражений

1. Пустой набор. $\{;\text{пустоеслово}\} = \emptyset$. В скобочной записи это же тождество имеет вид "равно(перечень(пустоеслово) пусто)".

2. Повторяющиеся элементы. Чтобы отбрасывать повторные вхождения элементов конечного списка, служит прием $\forall_{ab}(\{a, a; b\} = \{a; b\})$. Указатель "список(фикс(0 1 1))" означает, что набор "конкатенация(набор(x1 x1)x2)" идентифицируется без учета порядка элементов. Заменяющий терм обрабатывается нормализатором "нормперечень", завершающим процесс отбрасывания повторяющихся элементов. Последовательность действий откомпилированной программы такова: сначала в наборе выделяется произвольный элемент, рассматриваемый как первое вхождение a. Затем берется произвольное другое вхождение элемента того же набора, и проверяется совпадение его с первым вхождением. После этого строится заменяющий терм.

3. Объединение с перечнем. Если конечный список объединяется с некоторым множеством, то из него удаляются элементы, уже содержащиеся в данном множестве. Теорема приема такова: $\forall_{abc}(b \in a \to a \cup \{b; c\} = a \cup \{; c\})$. Антецедент обрабатывается проверочным оператором; указатель "список(фикс(0 1 2 1))" определяет идентификацию b с произвольным элементом набора. Указатель "модификатор" означает, что a будет идентифицироваться не с отдельным операндом объединения, а с объединением всех оставшихся операндов. При построении заменяющего терма используются нормализаторы "нормобъединение", "нормперечень".

4. Объединение перечней. $\forall_{ab}(\{; a\} \cup \{; b\} = \{; a; b\})$. Заменяющий терм обрабатывается нормализатором "нормперечень", устраняющим повторения элементов. Уровень срабатывания данного приема равен 0, в то время как предыдущего - 1 либо 3. Поэтому он будет перехватывать случаи объединения нескольких перечней, предыдущему же останутся лишь случаи объединения перечня с "неперечнем".

5. Пересечение с перечнем. $\forall_{abc}(\neg(b \in a) \to a \cap \{b; c\} = a \cap \{; c\})$. Отбрасываются элементы конечного списка, не входящие в пересекаемое с ним множество. В остальном - аналогично приему объединения с перечнем.

6. Вычитание перечня. $\forall_{abc}(\neg(b \in a) \to a \setminus \{b; c\} = a \setminus \{; c\})$. Аналогично предыдущему.

7. Вычитание из перечня. Рассматриваются два случая: элемент перечня принадлежит либо не принадлежит вычитаемому множеству.

$\forall_{abc}(b \in a \to \{b; c\} \setminus a = \{; c\} \setminus a)$

$\forall_{abc}(\neg(b \in a) \to \{b; c\} \setminus a = \{b\} \cup \{; c\} \setminus a)$

Первый прием аналогичен предыдущему; второй - несколько усилен. Уровень срабатывания его равен 4, и для обработки антецедента используется вспомогательная задача на доказательство. Он ориентирован на случаи, когда из перечня вычитается "не-перечень". Использование нормализатора "нормобъединение" позволяет сразу склеивать все выделяемые элементы b в общий список.

8. Разность перечней, имеющих общий элемент. $\forall_{abc}(\{b; a\} \setminus \{b; c\} = \{; a\} \setminus \{b; c\})$. Уровень срабатывания приема равен 0; заменяющий терм обрабатывается нормализатором "нормразность". Таким образом, вычитание конечных списков, имеющих хотя бы один общий элемент, будет выполняться за один шаг именно этим приемом.

Общая стандартизация утверждений

1. Принадлежность одноэлементному перечню. $\forall_{ab}(a = b \leftrightarrow a \in \{b\})$

   Тождество применяется справа налево.

2. Включение одноэлементного перечня.

   $\forall_{ab}(a \in b \leftrightarrow \{a\} \subseteq b)$.

   $\forall_{abc}(\neg(a \in b) \vee a \in c \leftrightarrow b \cap \{a\} \subseteq c)$

   В обоих случаях замена выполняется справа налево; во втором случае прием блокируется, если противоречит разрешению условия задачи на описание относительно неизвестных (либо оба выражения a, b известны, а c - не известно, либо c - неизвестная).

3. Непересечение с одноэлементным перечнем.

   $\forall_{ab}(\neg(a \in b) \leftrightarrow \text{непересек}(b, \{a\}))$

   $\forall_{abc}(\neg(a \in b) \vee \neg(a \in c) \leftrightarrow \text{непересек}(c, b \cap \{a\}))$

   Аналогично предыдущему.

4. Принадлежность перечню.

   $\forall_{abc}(a \in \{b; c\} \leftrightarrow a = b \vee a \in \{; c\})$

   Обычно прием применяется слева направо, хотя для специальных целевых установок, ориентированных на компактную переформулировку утверждений, предусмотрена версия, выполняющая встречное преобразование. Прием сопровождается большим количеством фильтров, регулирующих уровень его срабатывания (0,2,3 либо 6) и блокирующих нарушение стандартизации, определенной контекстом. Например, блокируется его применение в условиях задачи на описание, если a - неизвестная, а выражение под перечнем известно.

5. Непересечение с перечнем.

   $\forall_{abc}(\text{непересек}(c, \{a; b\}) \leftrightarrow \neg(a \in c)\ \&\ \text{непересек}(c, \{; b\}))$

   Этот и следующий приемы аналогичны предыдущему, однако количество фильтров у них существенно меньше.

6. Включение перечня.

   $\forall_{abc}(\{a;b\} \subseteq c \leftrightarrow \{;b\} \subseteq c \ \& \ a \in c)$

7. Непересечение множества и перечня, пересеченного с другим множеством.

   $\forall_{abcd}(\text{непересек}(d, a \cap \{b;c\}) \leftrightarrow (\neg(b \in a) \vee \neg(b \in d)) \ \& \ \text{непересек}(d, a \cap \{;c\}))$

   Здесь и далее предпринимается развертка гибридной конструкции, частью которой является перечень.

8. Включение пересечения множества с перечнем.

   $\forall_{abcd}(a \cap \{b;c\} \subseteq d \leftrightarrow (\neg(b \in a) \vee b \in d) \ \& \ a \cap \{;c\} \subseteq d)$

9. Непересечение множества и результата вычитания другого множества из перечня.

   $\forall_{abcd}(\text{непересек}(d, \{b;c\} \setminus a) \leftrightarrow (\neg(b \in d) \vee b \in a) \ \& \ \text{непересек}(d, \{;c\} \setminus a))$

10. Равенство двух перечней.

    $\forall_{ab}(\{a\} = \{b\} \leftrightarrow a = b)$

    $\forall_{abcd}(\{a,b\} = \{c,d\} \leftrightarrow a = c \ \& \ b = d \vee a = d \ \& \ b = c)$

    Направление замен - слева направо. Первая эквивалентность применяется на уровне 0; вторая - на уровне 2, причем преобразуемое равенство здесь является условием задачи на описание и содержит неизвестные.

11. Кванторная свертка в условие равенства одноэлементному перечню.

    $\forall_{ac}(a - \text{set} \rightarrow a = \{c\} \leftrightarrow \forall_b(b = c \leftrightarrow b \in a))$

    Эквивалентность применяется справа налево и лишь при отсутствии комментария "кванторнаясвертка", блокирующего исключение кванторов.

12. Существование одноэлементного перечня, равного данному множеству.

    $\forall_a(a - \text{set} \rightarrow \exists_b(a = \{b\}) \leftrightarrow \text{card} a = 1)$

13. Конъюнкция условий принадлежности перечню и отличия от элемента перечня.

    $\forall_{abx}(\neg(a \in \{;b\}) \rightarrow \neg(x = a) \ \& \ x \in \{a;b\} \leftrightarrow x \in \{;b\})$

14. Исключение антецедента, имеющего вид принадлежности перечню.

    $\forall_{ABan}(\forall_{xy}(x \in \{;\lambda_i(a(i), i \in \{1,\ldots,n\})\} \ \& \ A(x,y) \rightarrow B(x,y)) \leftrightarrow \forall_{iy}(i \in \{1,\ldots,n\} \ \& \ A(a(i),y) \rightarrow B(a(i),y)))$

    Здесь конечное семейство задано не явным перечислением элементов, а с помощью функциональной конструкции "отображение(...)", определяющей общий вид элемента семейства. Указатель "отображение(x1 x26 x27)" определяет идентификацию функциональных переменных a, A, B с произвольными термами; указатель "кортежпеременных(x24)" дает возможность идентифицировать y с остатком связывающей приставки квантора, имеющим произвольную (в том числе нулевую) длину. Указатель "внешнийквантор(фикс(0 1))" блокирует попытку извлечения квантора общности из квантора существования путем перехода к отрицаниям.

15. Равенство пересечения одноэлементному перечню.

    $\forall_{abcd}(d \in a \ \& \ d \in b \rightarrow (a \cup b) \cap c = \{d\} \leftrightarrow a \cap c = \{d\} \ \& \ b \cap c = \{d\})$

    Эквивалентность применяется слева направо; она позволяет перейти к более простым независимым условиям.

Свертка известных утверждений при редактировании ответа

Приведем серию приемов, используемых для компактной переформулировки сопровождающих утверждений при редактировании ответа задачи на описание. В каждом случае имеется фильтр, проверяющий отсутствие в преобразуемом утверждении неизвестных, а также фильтр, проверяющий наличие цели "соединение" (либо, в отдельных случаях, цели "свертка").

1. Свертка дизъюнкции двух равенств в принадлежность перечню.

   $\forall_{abc}(a = b \ \vee \ b = c \leftrightarrow b \in \{a, c\})$

   Если утверждение находится под квантором либо описателем, то данное преобразование может заблокировать другие важные приемы. Поэтому введены фильтры "свобоперанд(фикс(0 1 1))", "свобоперанд(фикс(0 1 2))".

2. Свертка дизъюнкции равенства и принадлежности перечню в принадлежность перечню.

   $\forall_{abc}(a \in \{b; c\} \leftrightarrow a = b \ \vee \ a \in \{; c\})$

3. Свертка конъюнкции принадлежности и отрицания равенства в принадлежность разности.

   $\forall_{abc}(\neg(b = c) \ \& \ b \in a \leftrightarrow b \in a \setminus \{c\})$

   Имеются две версии приема: одна для конъюнкции, расположенной в одном условии; другая - для двух различных условий. Во всех нижеследующих пунктах, относящихся к свертке конъюнкции, точно так же рассматриваются две версии.

4. Свертка конъюнкции включения и отрицания принадлежности во включение в разность.

   $\forall_{abc}(\neg(b \in c) \ \& \ c \subseteq a \leftrightarrow c \subseteq a \setminus \{b\})$

5. Свертка конъюнкции непересечения и отрицания принадлежности в непересечение с объединением.

   $\forall_{abc}(\text{непересек}(c, a \cup \{b\}) \leftrightarrow \neg(b \in c) \ \& \ \text{непересек}(a, c))$

6. Свертка конъюнкции принадлежности и включения во включение объединения.

   $\forall_{abc}(a \cup \{b\} \subseteq c \leftrightarrow a \subseteq c \ \& \ b \subseteq c)$

7. Свертка конъюнкции отрицаний равенства и принадлежности в отрицание принадлежности.

   $\forall_{abc}(\neg(b = c) \ \& \ \neg(b \in a) \leftrightarrow \neg(b \in a \cup \{c\}))$

8. Свертка конъюнкции отрицаний принадлежности в непересечение с перечнем.

   $\forall_{abc}(\text{непересек}(c, \{a, b\}) \leftrightarrow \neg(a \in c)\ \&\neg(b \in c))$

9. Свертка конъюнкции принадлежностей во включение перечня.

   $\forall_{abc}(\{a, b\} \subseteq c \leftrightarrow a \in c\ \&\ b \in c)$

10. Усмотрение принадлежности пересечению с перечнем.

    $\forall_{abcd}(b \in a \cap \{c; d\} \leftrightarrow b = c\ \&\ c \in a\ \vee\ b \in a \cap \{; d\})$

    Здесь и в остальных пунктах эквивалентность применяется справа налево.

11. Усмотрение принадлежности разности перечня и некоторого множества.

    $\forall_{abcd}(b \in \{c; d\} \setminus a \leftrightarrow \neg(c \in a)\ \&\ b = c\ \vee\ b \in \{; d\} \setminus a)$

12. Усмотрение включения пересечения с перечнем.

    $\forall_{abcd}(a \cap \{b; c\} \subseteq d \leftrightarrow \neg(b \in a \setminus d)\ \&\ a \cap \{; c\} \subseteq d)$

13. Усмотрение непересечения множества и пересечения другого множества с перечнем.

    $\forall_{abcd}(\text{непересек}(d, a \cap \{b; c\}) \leftrightarrow \neg(b \in a \cap d)\ \&\ \text{непересек}(d, a \cap \{; c\}))$

Группировка явных описаний для неизвестной

Несколько приемов, объединяющих два явно разрешенных относительно неизвестной утверждения в одно такое утверждение. Эти приемы обычно используются на завершающих этапах решения задачи, для склейки отдельных элементов ответа в одно утверждение. Кроме версии приема, относящейся к двум различным условиям задачи, обычно берется версия для конъюнкции тех же утверждений, расположенной в одном условии.

1. Два условия принадлежности.

   $\forall_{abc}(\{a, b\} \subseteq c \leftrightarrow a \in c\ \&\ b \in c)$

   Замена выполняется справа налево; c - неизвестная; a, b известны.

2. Дизъюнкция условий равенства и принадлежности.

   $\forall_{abc}(a \in \{b; c\} \leftrightarrow a = b\ \vee\ a \in \{; c\})$

   Указатель "дизъюнктоперанд" блокирует попытки применения приема к конъюнкции, рассматриваемой как отрицание дизъюнкции отрицаний, и выделения дизъюнкции из кванторной импликации. Фильтр "коммент(разборслучаев)" предотвращает устранение данным приемом дизъюнкции, специально введенной для разбора случаев.

3. Условия отрицания принадлежности и отрицания равенства.

   $\forall_{abc}(\neg(b = c)\ \&\ \neg(b \in a) \leftrightarrow \neg(b \in a \cup \{c\}))$

4. Условия включения и принадлежности.

   $\forall_{abc}(a \cup \{b\} \subseteq c \leftrightarrow a \subseteq c\ \&\ b \in c)$

5. Условия непересечения и отрицания принадлежности.

   $\forall_{abc}(\text{непересек}(c, a \cup \{b\}) \leftrightarrow \neg(b \in c) \ \& \ \text{непересек}(a, c))$

6. Условия включения и отрицания принадлежности.

   $\forall_{abc}(a \subseteq b \ \& \ \neg(c \in a) \leftrightarrow a \subseteq b \setminus \{c\})$

7. Условия принадлежности и отрицания равенства.

   $\forall_{abc}(\neg(b = c) \ \& \ b \in a \leftrightarrow b \in a \setminus \{c\})$

Усмотрение принадлежности перечню его элемента

$\forall_{ab}(a \in \{a; b\})$

Указатель "список(фикс(0 2 1))" обеспечивает поиск элемента, совпадающего с a, среди всех элементов набора под перечнем.

Усмотрение непустоты перечня

$\forall_{ab}(\neg(\{b; a\} = \emptyset))$

Прием заменяет на константу "ложь" равенство пустому множеству выражения, имеющего вид "перечень(набор(...))".

Подбор примера

$\forall_{ax}(x = \{a\} \rightarrow a \in x)$

Прием применяется к условию $a \in x$ задачи на описание, имеющей цель "пример". x - неизвестная задачи; a - выражение без неизвестных. Проверяется отсутствие содержащих x других условий задачи, кроме, быть может, условия "множество(x)". Указатель "подборзначений(1)" означает, что во вспомогательной задаче, создаваемой приемом, рассматриваемое условие будет заменено на условие $x = \{a\}$. Указатель "новый" необходим для контроля за изменениями задачи, после которых могут быть исключены прочие условия с неизвестной x.

Разбор случаев для условия принадлежности неизвестной перечню

Предусмотрен лишь прием для двухэлементного списка:

$\forall_{abc}(a = c \ \vee \ b = c \leftrightarrow c \in \{a, b\})$

Замена выполняется справа налево; c - неизвестная задачи на описание, не входящая ни в a, ни в b, но входящая в некоторое другое условие задачи. На этапе редактирования ответа прием блокируется. Измененное условие сопровождается комментарием "разборслучаев", приводящим к немедленному разбору случаев.

Нормализатор "нормперечень"

В нормализаторе имеется прием, устраняющий повторяющиеся элементы, а также прием, преобразующий перечень пустого набора в пустое множество.

4.12 Приемы символов "элемент" и "внешнийэлемент"

Чтобы выделять в непустом множестве A какой-то произвольный элемент, введено обозначение "элемент(A)". Аналогично, для обозначения некоторого элемента, не принадлежащего множеству A (считается, что такой элемент всегда существует), введено обозначение "внешнийэлемент(A)". Используются эти обозначения крайне редко, а так как почти никакой специальной информации об их значениях нет, то связанных с ними приемов немного.

Подбор примера

Чтобы указать в ответе задачи на поиск примера элемент заданного непустого множества, служит прием, основанный на следующей теореме:

$\forall_{ax}(a - \text{set} \ \& \ \neg(a = \emptyset) \ \& \ x = \text{элемент}(a) \rightarrow x \in a)$

Заголовок приема - "подборзначений". Утверждение $x \in a$ является условием задачи на описание, имеющей цель "пример". Переменная x - неизвестная, не встречающаяся в прочих условиях; выражение a не содержит неизвестных. Отсутствует цель "независит ...), ограничивающая зависимость неизвестных от параметров выражения a. В этой ситуации, согласно указателю "подборзначений(3)", прием пытается решить вспомогательную задачу, полученную из текущей задачи заменой рассматриваемого условия на $x = \text{элемент}(a)$. Предварительно проверяется истинность первых двух антецедентов.

Аналогичный прием имеется для символа "внешнийэлемент":

$\forall_{ax}(a - \text{set} \ \& \ x = \text{внешнийэлемент}(a) \rightarrow \neg(x \in a))$

Усмотрение принадлежности либо непринадлежности

Для усмотрения принадлежности значения "элемент(a)" непустому множеству a служит прием:

$\forall_{a}(a - \text{set} \ \& \ \neg(a = \emptyset) \rightarrow \text{элемент}(a) \in a)$

Для усмотрения непринадлежности множеству его "внешнего элемента" служит прием:

$\forall_{a}(\neg(\text{внешнийэлемент}(a) \in a))$

Оба приема имеют заголовок "второйтерм".

4.13 Приемы символа "прямоепроизведение"

В основном, рассматривавшиеся при обучении задачи содержали прямое произведение лишь двух сомножителей. Это отразилось и на представленных в разделе приемах. Впрочем, несколько приемов относятся к прямому произведению трех или четырех сомножителей, и несколько - к произвольному числу сомножителей. Напомним, что операция прямого произведения не ассоциативна. Обобщение приемов раздела на случай любого числа сомножителей можно рекомендовать в качестве хорошего самостоятельного упражнения, потребующего несложного развития компилятора ГЕНОЛОГа. Кроме операции прямого произведения, в разделе рассматривается операция "Проекция(A i)", значением которой служит проекция множества A наборов одной и той же длины n на i-ю координату $1 \leq i \leq n$.

Тождества общей стандартизации

1. Умножение на пустое множество

   $\forall_a(\emptyset \times a = \emptyset)$

   $\forall_a(a \times \emptyset = \emptyset)$

2. Прямое произведение одноэлементных множеств

   $\forall_{ab}(\{a\} \times \{b\} = \{(a, b)\})$

   Аналогичные приемы введены для трех, четырех и пяти сомножителей.

3. Пересечение прямых произведений

   $\forall_{abcd}((a \times c) \cap (b \times d) = (a \cap b) \times (c \cap d))$

4. Разность прямых произведений

   $\forall_{abc}((a \times c) \setminus (b \times c) = (a \setminus b) \times c)$

   $\forall_{abc}((c \times a) \setminus (c \times b) = c \times (a \setminus b))$

5. Объединение прямых произведений множеств с отброшенными точками

   $\forall_{abcd}(((a \setminus \{b\}) \times c) \cup (a \times (c \setminus \{d\})) = (a \times c) \setminus \{(b, d)\})$

6. Преобразование прямого произведения конечных списков в список пар

   $\forall_{abcde}(e = \{a; b\} \times \{c; d\} \to \{a; b\} \times \{c; d\} = e)$

   Прием обращается к нормализатору "нормпрямоепроизведение", который и выполняет указанное преобразование. Уровень срабатывания равен 3.

7. Усмотрение прямого произведения в множестве, заданном с помощью описателя "класс".

   $\forall_{AB}(\mathrm{set}_x(l(x) = 2 \ \& \ x - \text{слово} \ \& \ x(1) \in A \ \& \ x(2) \in B) = A \times B)$

   $\forall_{aB}(\mathrm{set}_x(l(x) = 2 \ \& \ x - \text{слово} \ \& \ x(1) = a \ \& \ x(2) \in B) = \{a\} \times B)$

   $\forall_{PQAB}(A = \mathrm{set}_y(P(y)) \ \& \ B = \mathrm{set}_y(Q(y)) \to \mathrm{set}_x(x - \text{слово} \ \& \ l(x) = 2 \ \& \ P(x(1)) \ \& \ Q(x(2))) = A \times B)$

   В последнем приеме выражения A, B обрабатываются вспомогательными задачами на упрощение.

Общая стандартизация утверждений

1. Пустота произведения множеств.

   $\forall_{ab}(a \times b = \emptyset \leftrightarrow a = \emptyset \ \vee \ b = \emptyset)$

   Эквивалентность применяется слева направо. В зависимости от контекста, уровень срабатывания приема варьируется от 0 до 3. Прием блокируется, если требуется получить сжатую формулировку условий.

2. Принадлежность произведению множеств.

   $\forall_{abf}(f \in a \times b \leftrightarrow \text{слово}(f) \ \& \ l(f) = 2 \ \& \ f(1) \in a \ \& \ f(2) \in b)$

   Эквивалентность применяется слева направо. Если преобразуется условие задачи на описание и f содержит неизвестные, то прием блокируется. Напомним, что пара f - функция, определенная на начальном отрезке натурального ряда от 1 до 2, так что $f(1), f(2)$ - первый и второй элементы пары. Уровень срабатывания приема равен 0, если преобразуемая принадлежность - посылка задачи. Иначе он больше 1. Если f имеет вид "набор(...)", то можно обойтись без ввода функциональных обозначений $f(\ldots)$:

$\forall_{abcd}((a,b) \in c \times d \leftrightarrow a \in c \ \& \ b \in d)$

Этот прием продублирован для наборов длины 3,4 и 5. Уровни срабатывания здесь равны 1. Если правая часть принадлежности не имеет вида прямого произведения, но некоторая посылка указывает, что она равна прямому произведению, то применяется другой прием:

$\forall_{abcddP}(P = c \times d \rightarrow (a,b) \in P \leftrightarrow a \in c \ \& \ b \in d)$

Введена его версия для трех сомножителей. В обоих случаях предполагается, что преобразуется условие задачи на описание, причем левая часть принадлежности содержит неизвестные. В качестве упражнения рекомендуем так модифицировать приемы, у которых правая часть принадлежности имеет вид прямого произведения, чтобы они могли срабатывать и во второй ситуации.

Еще два приема относятся к случаю, когда обе части принадлежности содержат неизвестные:

$\forall_{abf}(l(f) = 2 \ \& f(2) \in b \ \& \ f - \text{слово} \rightarrow f \in a \times b \leftrightarrow f(1) \in a)$

$\forall_{abf}(l(f) = 2 \ \& f(1) \in a \ \& \ f - \text{слово} \rightarrow f \in a \times b \leftrightarrow f(2) \in b)$

3. Непересечение произведений множеств.

   $\forall_{abcd}(\text{непересек}(a \times b, c \times d) \leftrightarrow \text{непересек}(a,c) \ \vee \ \text{непересек}(b,d))$

   Так как прием преобразует утверждения указанного вида в дизъюнкции, то для предотвращения излишнего разбора случаев его применение к корневым вхождениям в посылки блокируется. Однако, для посылок задач на доказательство введена отдельная версия приема, срабатывающая на более высоком уровне.

4. Включение произведений множеств.

   $\forall_{abcd}(a \times b \subseteq c \times d \leftrightarrow a \subseteq c \ \& \ b \subseteq d \ \vee \ a = \emptyset \ \vee \ b = \emptyset)$

   Аналогично предыдущему.

5. Равенство произведений множеств.

   $\forall_{abcd}(a \times b = c \times d \leftrightarrow (a = \emptyset \ \vee \ b = \emptyset) \ \& \ (c = \emptyset \ \vee \ d = \emptyset) \ \vee \ a = c \ \& \ b = d)$

6. Преобразование условия принадлежности прямому произведению перечней в дизъюнкцию равенств.

   Если перемножаются два конечных списка, то условие принадлежности этому произведению преобразуется в дизъюнкцию равенств парам, образованным элементами списков. Прием основан на следующей теореме:

   $\forall_{abmnx}(x \in \{;\lambda_i(a(i), i \in \{1,\ldots,n\})\} \times \{;\lambda_j(b(j), j \in \{1,\ldots,m\})\} \leftrightarrow \exists_{ij}(i \in \{1,\ldots,n\} \ \& \ j \in \{1,\ldots,m\} \ \& \ x = (a(i), b(j))))$

   Указатель "развертка(фикс(0 1 2 1 1)фикс(0 1 2 2 1))" выделяет вхождения подвыражений $\lambda_i(a(i), i \in \{1,\ldots,n\})$, $\lambda_j(b(j), j \in \{1,\ldots,m\})$. Это означает, что подвыражения будут идентифицироваться с термами "набор($a(1)\ \dots\ a(n)$)", "набор($b(1)\ \dots\ b(m)$)", по которым и определяются "функции" a, b. Указатель "или(фикс(0 2)фикс(0 2 3 1)фикс(0 2 3 2))" относится к квантору существования и определяет развертку его в дизъюнкцию. Указатели вхождений "фикс(0 2 3 1)", "фикс(0 2 3 2)" выделяют утверждения $i \in \{1, \dots, n\}$, $j \in \{1, \dots, m\}$, перечисляющие при развертке значения связанных переменных квантора. Фильтры ограничивают срабатывание приема случаями, когда x - неизвестная задачи на описание, встречающаяся в каких-либо других условиях. Тогда бывает полезным просмотр возможных значений x и отсечение лишних значений.

7. Принадлежность прямому произведению, сомножителем которого является одноэлементный перечень

   $\forall_{abc}(a \in \{b\} \times c \leftrightarrow \exists_d(d \in c \ \& \ a = (b, d)))$

   Замена выполняется слева направо. Прием используется в тех случаях, когда a - либо неизвестная задачи на описание, либо параметр редактируемого параметрического описания. Предполагается, что заголовком выражения c не является символ "перечень" и что имеется другое условие, содержащее a. Преобразованное условие, имеющее своим заголовком квантор существования, снабжается комментариями "серия", "фильтрсерии". Они обеспечат переход к вспомогательной задаче, у которой квантор по d будет отброшен и станет возможной подстановка пары (b, d) в остальные содержащие a условия.

   Кроме указанного приема, имеется также его двойник, содержащий перечень во втором сомножителе.

8. Принадлежность прямому произведению переменного числа одноэлементных перечней.

   Чтобы задавать прямое произведение набора множеств A, в логическом языке введено обозначение "Прямоепроизведение(A)". Как обычно, общий вид набора A определяется при помощи конструкции "отображение(i и(целое(i)меньшеилиравно(1 i) меньшеилиравно(i n)) $A(i)$)". На экране прямое произведение набора переменной длины прорисовывается так же, как обычное конечное произведение.

$$\forall_{Abn}(b \in \prod_{i=1}^{n}\{A(i)\} \leftrightarrow b = \lambda_i(A(i), i \in \{1, \dots, n\}))$$

9. Принадлежность прямому произведению переменного числа одинаковых множителей.

$$\forall_{xAn}(x \in \prod_{i=1}^{n} A \leftrightarrow \text{кортеж}(x, n, A))$$

10. Принадлежность произведению пересечения.

    $\forall_{bcef}(f(1) \in c \rightarrow f \in (b \cap c) \times e \leftrightarrow f \in b \times e)$

    Аналогичный прием имеется для переставленных сомножителей.

11. Непересечение с произведением объединения.

    $\forall_{bdfg}(\text{непересек}(d, f \times b)\ \&\ \text{непересек}(d, g \times b) \leftrightarrow \text{непересек}(d, (f \cup g) \times b))$

    Замена выполняется справа налево; блокируется преобразование утверждений, явно разрешенных относительно неизвестной. Аналогичный прием имеется для переставленных сомножителей.

Свертка известных утверждений при редактировании ответа

В этом разделе имеется несколько простых приемов, ориентированных на сокращенную переформулировку сопровождающих утверждений ответа задачи на описание. Они обратны приемам, применявшимся на основном этапе решения задачи: дизъюнкция двух условий непересечения преобразуется в условие непересечения прямых произведений; восстанавливается условие принадлежности прямому произведению, и т.п.

Кванторные свертки с участием произведения множеств

Приводимые ниже приемы позволяют исключать кванторы. Они срабатывают при отсутствии комментария "кванторнаясвертка". Кроме того, приемы блокируются при редактировании ответа задач на описание, имеющих цель "попыткапараметризации". В первых трех пунктах преобразование применяется справа налево, далее - слева направо.

1. Равенство множества прямому произведению двух множеств.

   $\forall_{abc}(c{-}\text{set} \rightarrow c = a \times b \leftrightarrow \forall_f(l(f) = 2\ \&\ f(1) \in a\ \&\ f(2) \in b\ \&\ f{-}\text{слово} \leftrightarrow f \in c))$

2. Включение прямого произведения.

   $\forall_{abc}(a \times b \subseteq c \leftrightarrow \forall_f(l(f) = 2\ \&\ f(1) \in a\ \&\ f(2) \in b\ \&\ f - \text{слово} \rightarrow f \in c))$

3. Непересечение с прямым произведением.

   $\forall_{abc}(\text{непересек}(c, a \times b) \leftrightarrow \forall_f(l(f) = 2\ \&\ f(1) \in a\ \&\ f(2) \in b\ \&\ f{-}\text{слово} \rightarrow \neg(f \in c)))$

4. Усмотрение включения в подпроизведение.

   $\forall_{ABCa}(C \subseteq A \times B \rightarrow \forall_x(x \in C \rightarrow x(1) = a) \leftrightarrow C \subseteq \{a\} \times B)$

   $\forall_{ABC}(C \subseteq A \times B \rightarrow \forall_{ab}(a \in C\ \&\ b \in C \rightarrow a(1) = b(1)) \leftrightarrow \exists_d(d \in A\ \&\ C \subseteq \{d\} \times B))$

   Аналогичные приемы имеются и для вторых координат.

Подбор примера

Чтобы подбирать пример множества, произведение которого на другое дает пустое множество, введены два приема:

$\forall_{ax}(x = \emptyset \rightarrow a \times x = \emptyset)$

$\forall_{ax}(x = \emptyset \rightarrow x \times a = \emptyset)$

Преобразование выражений с неизвестными

1. Объединение прямых произведений.

   $\forall_{abc}((a \times c) \cup (b \times c) = (a \cup b) \times c)$

   Прием выполняет группировку прямых произведений для неизвестного выражения c относительно известных a, b. Аналогичный прием введен для случая, когда первый сомножитель не известен.

2. Разность прямых произведений.

   $\forall_{abc}((c \times a) \setminus (b \times a) = (c \setminus b) \times a)$

   a содержит неизвестные; b, c - не содержат. Аналогичный прием введен для первого сомножителя.

Попытка представить в виде прямого произведения подвыражение уравнения

Преобразование множества к виду прямого произведения может позволить разрешить относительно неизвестных условие задачи на описание, уже содержащее прямое произведение. Для такого преобразования применяется нормализатор "видпрямоепроизведение".

1. Противоположная часть уравнения имеет вид прямого произведения.

   $\forall_{acde}(a = e \rightarrow e = c \times d \leftrightarrow a = c \times d)$

   Замена применяется слева направо; переменной a присваивается результат обработки выражения e нормализатором "видпрямоепроизведение". Равенство входит в условие задачи на описание и содержит неизвестные. Выражение e отлично от переменной и не имеет заголовка "прямоепроизведение"; выражение a либо имеет заголовок "прямоепроизведение", либо короче выражения e.

2. Противоположная часть включения имеет вид прямого произведения.

   $\forall_{acde}(a = e \rightarrow e \subseteq c \times d \leftrightarrow a \subseteq c \times d)$

   Аналогичный прием - для преобразования правой части включения.

3. Противоположная часть условия непересечения имеет вид прямого произведения.

   $\forall_{acde}(a = e \rightarrow \text{непересек}(e, c \times d) \leftrightarrow \text{непересек}(a, c \times d))$

Попытка преобразования к виду объединения пересечений

Для прямого произведения, как и для ранее рассмотренных теоретико-множественных операций, предусмотрена попытка упрощения выражений путем преобразования их к стандартной форме "объединение пересечений". Теорема приема имеет вид:

$\forall_{abc}(c = a \times b \rightarrow a \times b = c)$

Переменной c присваивается результат последовательной обоаботки прямого произведения $a \times b$ нормализаторами "стандобъединение" (развертка выражения в объединение пересечений и упрощение), "группмножество", "свертка" (обратная свертка выражения). Проверяется, что результат короче исходного выражения. Чтобы попытка упрощения применялась только к максимальным подвыражениям, проверяется отличие внешней операции от символов "объединение", "пересечение", "разность", "прямоепроизведение".

Определение неизвестной пары

Если найдены обе компоненты неизвестной пары, то сборка их в результирующую пару выполняется следующим приемом:

$\forall_{abx}(l(x) = 2 \to x(1) = a \;\&\; x(2) = b \leftrightarrow x = (a, b))$

Здесь x - неизвестная задачи на описание; a, b - известны. Заголовок приема - "заменаусловия(второйтерм)".

Проекция множества наборов

Для операции "Проекция(...)" приведены лишь несколько простых приемов, позволяющих находить проекцию конечного списка пар.

Нормализатор общей стандартизации

Нормализатор "нормпрямоепроизведение" исключает умножение на пустое множество и преобразует произведение двух конечных списков в список пар.

Нормализатор "видпрямоепроизведение"

1. Объединение произведений.

   $\forall_{abc}((b \times a) \cup (b \times c) = b \times (a \cup c))$

   Указатель "набор(второйтерм)" означает, что рассматриваются все члены объединения, и преобразование выполняется, если их первые множители совпадают. Аналогичный прием имеется для вторых множителей.

2. Пересечение произведений.

   $\forall_{abcd}((a \times c) \cap (b \times d) = (a \cap b) \times (c \cap d))$

   Указатель "модификатор" означает, что преобразование применяется лишь к пересечению двух множеств.

3. Попытка преобразования группы операндов пересечения к виду произведения.

   Если пересечение имеет одним из своих операндов прямое произведение, то применяется прием, предпринимающий попытку преобразовать к виду произведения группу остальных операндов. Таким образом подготавливается возможность применения предыдущего приема.

   $\forall_{abde}(a = e \to e \cap (b \times d) = a \cap (b \times d))$

   Переменной a присваивается результат обработки операнда e нормализатором "видпрямоепроизведение". Предполагается, что выражение e содержит неизвестные, отлично от переменной и не имеет заголовка "прямоепроизведение". Оно идентифицируется с пересечением всех операндов, отличных от выделенного операнда $b \times d$.

4. Разность произведений.

   $\forall_{abc}((c \times a) \setminus (b \times a) = (c \setminus b) \times a)$

   Аналогичный прием - для совпадающих первых операндов разностей.

4.14 Приемы, связанные с семействами множеств

Напомним обозначения, используемые в системе для работы с семействами множеств. Утверждение "семействомножеств(f)" означает, что f есть функция, значения которой суть множества. Для обозначения объединения и пересечения множеств семейства f служат выражения "объединениевсех(f)", "пересечениевсех(f)". Так как обычно семейства множеств задаются с помощью описателя "отображение", в теоремах приемов применяются записи "объединениевсех($x_1 \ldots x_n$ $P(x_1 \ldots x_n)$ $f(x_1 \ldots x_n)$)" и "пересечениевсех($x_1 \ldots x_n$ $P(x_1 \ldots x_n)$ $f(x_1 \ldots x_n)$)". Они позволяют идентифицировать выражение $f(\ldots)$ и утверждение $P(\ldots)$, определяющие семейство. Функциональные переменные f, P выделяются при этом указателем "отображение". Формульный редактор прорисовывает данные записи в виде:

$$\bigcup_{x_1,\ldots,x_n,P(x_1\ldots x_n)} f(x_1 \ldots x_n), \qquad \bigcap_{x_1,\ldots,x_n,P(x_1\ldots x_n)} f(x_1 \ldots x_n).$$

Если $n = 1$ и $P(i)$ имеет вид "и(целое(i)меньшеилиравно(1 i)меньшеилиравно(i n))", то прорисовка приобретает вид:

$$\bigcup_{i=1}^{n} f(i), \quad \bigcap_{i=1}^{n} f(i).$$

Утверждения "убывмножества(f)", "возрастмножества(f)" означают, что f есть последовательность множеств, нестрого убывающая либо, соответственно, возрастающая по включению.

При обучении решателя теории вероятностей возникла необходиость в следующих обозначениях. Утверждение "семействоэлементов(f g)" означает, что f есть семейство множеств, а g - функция, область определения которой можно так взаимнооднозначно отобразить на область определения семейства f, что каждое значение g будет принадлежать соответствующему множеству семейства f. Утверждение "Непересек(A f)" означает, что множество A не пересекается ни с одним из элементов семейства множеств f. Выражение "слойсемейства(f A n)" обозначает множество всех таких элементов множества A, которые принадлежат в точности n множествам семейства f. Здесь n - целое неотрицательное. Если $F = (f_1, \ldots, f_n)$ - набор семейств множеств, имеющих общую область определения и для каждого элемента этой области образующих n попарно непересекающихся подмножеств множества A; $m = (m_1, \ldots, m_n)$ - набор целых неотрицательных чисел, - то выражение "слойсемейств(F A m)" обозначает множество всех элементов множества A, принадлежащих ровно m_1 множествам семейства f_1, ровно m_2 множествам семейства f_2, и т.д. вплоть до f_n. Утверждение "разбиения(f)" означает, что f есть семейство семейств множеств, имеющих одну и ту же область определения, причем для каждого элемента этой области соответствующие множества семейств попарно не пересекаются. Мощность множества элементов семейства множеств f, которым принадлежит объект a, обозначается "числопопаданий(a f)".

Наконец, в данном разделе рассматривается также понятие разбиения множества, хотя оно и формализовано не через семейства, а через множества множеств. Именно, утверждение "разбиение(A B)" означает, что B есть множество непересекающихся подмножеств множества A, объединение которых равно A.

Приемы символа "семействомножеств"

1. Кванторная свертка

   $\forall_{fg}(\text{семействомножеств}(\lambda_x(f(x), g(x))) \leftrightarrow \forall_x(g(x) \rightarrow f(x) - \text{set}))$

   Преобразование применяется справа налево. Кванторная импликация входит в условие задачи на описание, не имеющей комментария "кванторнаясвертка". Уровень срабатывания равен 2.

2. Кванторная расшифровка

   Теорема приема та же, что в предыдущем пункте, но направление замены - слева направо. Уровень срабатывания - от 6 до 8, в зависимости от контекста. Преобразование блокируется, если утверждение расположено под описателями "класс", "отображение". Прием вводит комментарий "кванторнаясвертка", блокирующий обратную замену.

3. Усмотрение множества

   $\forall_{fgx}(\text{семействомножеств}(\lambda_a(f(a), g(a))) \ \& \ g(x) \rightarrow f(x) - \text{set})$

   Явное указанное в контексте утверждение "семействомножеств(...)" используется для усмотрение истинности утверждения "множество(...)". Последнее заменяется на константу "истина". Второй антецедент обрабатывается проверочными операторами. Заметим, что его заголовок заранее не известен, и компилятор не может обеспечить вызов конкретного проверочного оператора. Поэтому антецедент выделен не указателем "блокпроверок(...)", а указателем "очевидно(...)". Компилятор организует обращение к оператору "очевидно(...)", который анализирует заголовок конкретного проверяемого утверждения и переадресует его соответствующему проверочному оператору.

4. Усмотрение семейства множеств

   $\forall_A(\text{семействомножеств}(A) \rightarrow \text{семействомножеств}(A))$

   Прием организует обращение к проверочному оператору "усмсемействомножеств" для проверки утверждения "семействомножеств(...)". При успехе это утверждение заменяется на константу "истина". Прием применяется только к утверждениям, связанным внешними кванторами и описателями, так как прочие вхождения могут быть полезны как сопровождающая информация.

5. Константное семейство

   $\forall_{af}(\text{семействомножеств}(\lambda_x(a, f(x))) \leftrightarrow a - \text{set})$

   Прием общей стандартизации.

6. Пустое семейство

   $\forall_f(\text{семействомножеств}(\lambda_x(f(x), \text{ложь}))$

   $\forall_f(f - \text{функция} \ \& \ \text{Dom}(f) = \emptyset \rightarrow \text{семействомножеств}(f))$

Оба приема усматривают семейство множеств в функции с пустой областью определения. Во втором приеме первый антецедент обрабатывается проверочным оператором, а второй идентифицируется с утверждением контекста.

7. Проверочный оператор "усмсемействомножеств".

   Оператор имеет всего два специальных приема, основанных на теоремах:

   $\forall_{AP}(\text{семействомножеств}(A) \rightarrow \text{семействомножеств}(\lambda_i(A(i), P(i))))$

   $\forall_{AP}(A(i) - \text{set} \rightarrow \text{семействомножеств}(\lambda_i(A(i), P(i))))$

   В первом случае антецедент непосредственно присутствует в контексте. Здесь A - обычная переменная; $P(i)$ - функциональная переменная, идентифицируемая с произвольным утверждением, ограничивающим область определения семейства A. Во втором случае $A(i), P(i)$ - функциональные переменные, идентифицируемые с произвольными термами. Антецедент обрабатывается проверочным оператором "усммножество".

Приемы символа "объединениевсех"

1. Кванторная свертка в условие принадлежности объединению семейства множеств. Этот прием применяется в задачах на описание, у которых либо есть цель, определяющая тенденцию к получению бескванторного утверждения (безотносительно к возможному появлению новых описателей), либо в результате свертки происходит явное разрешение относительно неизвестной. Теорема приема имеет вид:

   $\forall_{afg}(a \in \bigcup_{x,f(x)} g(x) \leftrightarrow \exists_x(a \in g(x) \ \& \ f(x)))$

   Направление преобразования - справа налево. Уровень срабатывания равен 2. Переменные f, g выделены указателем "отображение". Указатель "кванторнаясвертка" распространяет действие приема и на кванторы общности, представляемые при идентификации как отрицание квантора существования. Прием блокируется комментарием "кванторнаясвертка".

2. Кванторная расшифровка условия принадлежности объединению семейства множеств. Прием основан на той же самой теореме, но направление преобразования - слева направо. Текущая задача - на доказательство либо на описание, причем в последнем случае выражение "объединениесемейства(...)" должно либо находиться в посылках, либо содержать неизвестные. Уровень срабатывания приема равен 3; после его применения вводится комментарий "кванторнаясвертка", блокирующий обратные переходы.

3. Объединение константного семейства.

   $\forall_{af}(\neg(\text{set}_x(f(x)) = \emptyset) \ \& \ a - \text{set} \rightarrow \bigcup_{x,f(x)} a = a)$

   Первый антецедент обычно выполняется, и поэтому для его проверки используется немного усиленное средство - вспомогательная задача на доказательство, решаемая до уровня 4. Такое усиление, возможно, чуть-чуть замедляет применение преобразования, зато позволяет выполнить его, если непустота семейства не совсем очевидна. Второй антецедент обрабатывается проверочным оператором.

4. Объединение пустого семейства.

$$\forall_f(\bigcup_{x,\text{ложь}} f(x)=\emptyset)$$

5. Объединение одноэлементного семейства.

$$\forall_{af}(\bigcup_{i,i=a} f(i)=f(a))$$

6. Объединение семейства объединений.

$$\forall_{fgh}(\bigcup_{x,h(x)} f(x)\cup\bigcup_{x,h(x)} g(x)=\bigcup_{x,h(x)}(f(x)\cup g(x)))$$

   Преобразование применяется справа налево - переходим к более простым описателям. Прием имеет уровень срабатывания 1 и применяется в условиях задач на преобразование. Комментарий "длина", указывающий на завершающую обработку ответа, блокирует замену.

7. Объединение семейства пересечений с константным выражением.

$$\forall_{afg}(a\cap\bigcup_{x,g(x)} f(x)=\bigcup_{x,g(x)}(a\cap f(x)))$$

   Преобразование выполняет общую стандартизацию и применяется справа налево; уровень срабатывания равен 1.

8. Объединение семейства разностей с константным вторым операндом.

$$\forall_{afg}(\bigcup_{x,g(x)} f(x)\setminus a=\bigcup_{x,g(x)}(f(x)\setminus a))$$

   Аналогично предыдущему.

9. Объединение семейства разностей с константным первым операндом.

$$\forall_{afg}(a\setminus\bigcap_{x,g(x)} f(x)=\bigcup_{x,g(x)}(a\setminus f(x)))$$

10. Объединение двух подсемейств одного семейства.

   Это преобразование, хотя и похоже на приведенное выше преобразования объединения семейства объединений, применяется в противоположном направлении. Для него имеются два приема, основанные на следующих теоремах:

$$\forall_{fgh}(\bigcup_{i,g(i)} f(i)\cup\bigcup_{j,h(j)} f(j)=\bigcup_{i,g(i)\vee h(i)} f(i))$$

$$\forall_{fgas}(i-\text{число}\ \&\ (i-\text{число}\ \&\ a=f(i))=(i=s)\to\bigcup_{i,g(i)} f(i)\cup a=\bigcup_{i,g(i)\vee i=s} f(i))$$

Направления замены в обоих случаях - слева направо. Первый прием объединяет в общее семейство два семейства, элементы которых определяются идентичными выражениями $f(\ldots)$; второй - усматривает, что дополнительное множество a может быть представлено как $f(s)$. Первый антецедент обрабатывается проверочным оператором, к посылкам которого добавляется утверждение $g(i)$. Он проверяет, что условие g допускает только числовые значения параметра i. Второй антецедент выделен указателем "идентификатор". Его левая часть - уравнения "i – число, $a = f(i)$" для определения указанного значения s. Она обрабатывается вспомогательной задачей на описание, задаваемой нормализатором "задача(4 тип(описать) полный явное прямойответ упростить цель(неизвестная(x9)))". Если ответ вспомогательной задачи имеет вид равенства $i = s$, то он идентифицирует значение s. Во избежание чрезмерно долгих попыток решения задачи, прием снабжен указателем "лимит(200000)".

11. Поглощение для двух объединений семейств.

$$\forall_{ABP}(\bigcup_{i,P(i)} A(i) \cap \bigcup_{j,P(j)} (A(j) \setminus B(j)) = \bigcup_{j,P(j)} (A(j) \setminus B(j)))$$

12. Объединение по одноэлементным подмножествам.

$$\forall_{aPf}(\bigcup_{i,\mathrm{card}\, i=1, i\subseteq a, P(i)} f(i) = \bigcup_{i, i\in a, P(\{i\})} f(\{i\}))$$

Преобразование применяется слева направо и позволяет упростить индексацию семейства.

13. Вынесение отдельного члена объединения.

$$\forall_{aPf}(Q(a) \to \bigcup_{i,i=a\vee P(i),Q(i)} f(i) = f(a) \cup \bigcup_{i,P(i),Q(i)} f(i))$$

Преобразование применяется слева направо; проверяется, что $P(i)$ пусто либо представляет собой дизъюнкцию равенств, фиксирующих значения индекса i. Антецедент выделен указателем "очевидно", задающим использование проверочных операторов. Аналогичный прием основан на теореме:

$$\forall_{abPf}(P(a) \to \bigcup_{i,i\in\{a;b\},P(i)} f(i) = f(a) \cup \bigcup_{i,i\in\{;b\},P(i)} f(i))$$

14. Развертка конечного объединения.

Если в посылках задачи на исследование встречается объединение семейства A, распространяемое на фиксированное конечное число членов, причем прочие посылки содержат выражения $A(t)$, не связанные внешними кванторами и описателями, то предпринимается переход к обычному объединению. Теорема приема имеет вид:

$$\forall_{ABmnk}(k = n - m + 1 \to \bigcup_{i=m}^{n} A(i) = B \leftrightarrow B = \bigcup_{j=1}^{k} A(m+j-1))$$

Фильтры "целое(x13)", "целое(x14)" определяют идентификацию m, n с десятичными записями целых чисел. Антецедент определяет число k объединяемых членов. Он выделен указателем "программа", так как его правая часть допускает непосредственное вычисление. Фильтр "меньше(x11 8)" задает верхнюю границу для k, равную 8. Указатель "развертка(фикс(0 2 2))" означает, что объединение семейства в правой части будет развернуто в обычное объединение нескольких множеств.

15. Определение объединения множеств с помощью вспомогательной задачи на описание.

    Чтобы упростить выражение для объединения семейства, часто бывает полезно разрешить условие принадлежности объединению относительно индекса семейства. Имеются две версии такого приема, основанные на следующих теоремах:

$$\forall_{PQR}(\exists_x(P(x)\ \&\ Q(x,y)) = R(y) \to \bigcup_{x,P(x)} \mathrm{set}_y(Q(x,y)) = \mathrm{set}_y(R(y)))$$

$$\forall_{PQR}(\exists_x(P(x)\ \&\ y \in Q(x)) = R(y) \to \bigcup_{x,P(x)} Q(x) = \mathrm{set}_y(R(y)))$$

    В первом случае общий элемент семейства задается через описатель "класс"; здесь вероятность получить упрощение при разрешении относительно индекса семейства больше, и уровень срабатывания приема равен 3. Во втором случае выражение для общего элемента семейства не имеет заголовка "класс", и уровень срабатывания выше - он равен 6. Антецеденты в обоих случаях имеют вид равенства, левая часть которого представляет собой условие принадлежности элемента y объединению семейства. Подкванторное утверждение этого условия обрабатывается нормализатором - вспомогательной задачей на описание, имеющей своей неизвестной индекс семейства x. Ответ задачи подставляется под квантор существования и обрабатывается еще одной задачей, на этот раз имеющей тип "преобразовать". Ее максимальный уровень невелик (равен 4), так что происходит лишь общая стандартизация утверждения. После выполнения указанных действий идентифицируется функциональная переменная $R(y)$, и далее формируется заменяющее выражение $\mathrm{set}_y(R(y))$. Оно также обрабатывается нормализаторами. Первый прием применяется без дополнительных ограничений; второй - лишь при условии,что результирующее выражение не содержит описателя "класс".

16. Нормализатор общей стандартизации "нормобъединениевсех".

    В этом нормализаторе собраны лишь немногие из приведенных выше преобразований: объединение константного семейства; вынесение отдельного члена объединения; объединение по одноэлементным подножествам, и объединение одноэлементного семейства.

Приемы символа "пересечениевсех"

Эти приемы аналогичны приемам предыдущего раздела. Рекомендуется самостоятельно ознакомиться с ними по оглавлению базы приемов.

Приемы символа "разбиение"

Представленные в разделе приемы возникли при рассмотрении нескольких задач по теории вероятностей и комбинаторике. Их список не претендует на какую-либо полноту. Систематического обучения системы решению задач с разбиениями не предпринималось. При первом чтении данный раздел можно опустить. В любом случае, рекомендуем найти в разделах "Теория вероятностей" и "Теория множеств" те задачи, где данные приемы используются. Напомним последовательность необходимых для этого действий. Войдя в просмотр приема, следует нажать "shift-3" (переход в оглавление задачника); выделить в задачнике соответствующий раздел; нажать "shift-2" (обратный переход в оглавление базы приемов); снова войти в просмотр приема, и нажать "shift-A" (кириллица). После небольшой паузы, необходимой системе для поиска в файлах, нажать "shift-3" и снова оказаться в оглавлении задачника. Далее нажатие клавиши "ф" переводит в буфер задачника, где создан подраздел с тем же названием, что у текущего пункта оглавления базы приемов. В нем перечислены все задачи выбранного раздела задачника, при решении которых прием срабатывал.

1. Мощность разбиения на равномощные подмножества.

   $\forall_{ABmn}(\text{разбиение}(A,B)\ \&\ \text{card}B = m\ \&\ m - \text{число}\ \&\ \forall_i(i \in B \rightarrow \text{card}i = n)\ \&\ n - \text{число} \rightarrow \text{card}A = mn)$

   Посредством $cardA$ обозначаем мощность множества A. Прием выполняет вывод следствий в посылках. Первый и четвертый антецеденты, указывающие, что имеется разбиение B множества A и что каждый класс разбиения содержит ровно n элементов, непосредственно идентифицируются в посылках. Второй антецедент присваивает переменной m выражение для числа классов разбиения B, обработанное нормализатором "нормощность". Последний нормализатор будет рассматриваться нами в комбинаторике; он содержит лишь самые простые приемы для непосредственного усмотрения мощности.

2. Усмотрение разбиения из мощностных соображений.

   Если некоторое конечное множество A представлено в виде объединения подмножеств, причем мощность его равна сумме мощностей подмножеств, то подмножества не пересекаются и образуют разбиение. В этой ситуации срабатывает прием, заменяющий утверждение о равенстве множества A объединению подмножеств на утверждение о том, что эти подмножества образуют разбиение.

   $\forall_{Aym}(\text{конечное}(A)\ \&\ m = l(y)\ \&\ \sum_{i=1}^{m} \text{card}y(i) = cardA \rightarrow \bigcup_{i=1}^{m} y(i) = A \leftrightarrow \text{разбиение}(A, \{;y\}))$

   Первый антецедент обрабатывается проверочным оператором; второй и третий - выделены указателем "идентификатор". При сравнении мощности множества A с суммой мощностей подмножеств используется нормализатор "нормощность". Сумма мощностей обрабатывается нормализатором "нормсуммавсех", который будет подробнее рассмотрен в разделе "Элементарная алгебра".

3. Перестановка классов разбиения.

   $\forall_{xnAB}(\text{разбиение}(B,A) \rightarrow \text{кортеж}(x,n,A)\ \&\ \text{разбиение}(B,\{;x\}) \leftrightarrow \text{перестановка}(x,A))$

   Набор классов разбиения сам образует разбиение, если он является перестановкой данных классов. Замена выполняется слева направо. Антецедент обрабатывается вспомогательной задачей на доказательство.

4. Перечисление разбиений, составленных из заданных подмножеств.

   Прием позволяет находить все разбиения множества, составленные из элементов заданного конечного списка его подмножеств. Фактически он инициирует разбор случаев, приводящий к перебору систем подмножеств. Так как при этом срабатывают различные приемы отсечения нереализуемых вариантов, перебор в разумных случаях требует не очень большого времени.

   $\forall_{AxabPQ}(a \subseteq A \,\&\, (\text{разбиение}(A \backslash a, y) \,\&\, y \subseteq \{;b\}) = P(y) \,\&\, (\text{разбиение}(A, x) \,\&\, x \subseteq \{;b\}) = Q(x) \rightarrow \text{разбиение}(A, x) \;\&\; x \subseteq \{a;b\} \leftrightarrow Q(x) \;\vee\; \exists_y(P(y) \;\&\; x = y \cup \{a\}))$

   Заголовок приема "заменаусловия(второйтерм)" означает, что выполняется замена условий "разбиение(A, x)", "$x \subseteq \{a;b\}$" задачи на описание. Переменная x является неизвестной; выражения A, a, b известны. Чтобы элемент a списка подмножеств, из которых требуется составить разбиение, идентифицировался с произвольным (не обязательно первым) элементом, введен указатель "список(фикс(0 1 2 2 1))". Первый антецедент проверяет, что выбраныый элемент является подмножеством множества A; для этого применяется проверочный оператор "усмсодержится". Второй и третий антецеденты представляют собой обращения к вспомогательным задачам. Первая из них находит все разбиения y множества $A \setminus a$, составленные из оставшихся после удаления a элементов списка; вторая - все разбиения x множества A, полученные без использования a. Соответственно, возникают утверждения $P(y)$ и $Q(x)$, определяющие эти разбиения в явном виде (например, в виде дизъюнкции или условия принадлежности конечному перечню). Заменяющая часть представляет собой дизъюнкцию условия $Q(x)$, охватывающего все разбиения без a, и условия, выражающего разбиения с a через разбиения y множества $A \setminus a$. Прием вводит комментарий (разбиение перечисление A), блокирующий повторные попытки применения. Это важно, чтобы отсечь разбор случаев по другим версиям элемента a. Однако, такое отсечение должно выполняться лишь после проверки истинности первого антецедента. Поэтому указатель "ключ(разбиение перечисление x26 посылки(1))", имеет служебный терм "посылки(1)" задающий номер антецедента, после обработки которого будет введен комментарий.

5. Усмотрение нарушения требований разбиения.

   $\forall_{Axa}(\neg(A \subseteq a) \;\&\; x \subseteq \{a\} \rightarrow \neg(\text{разбиение}(A, x)))$

   $\forall_{A}(\neg(A = \emptyset) \rightarrow \neg(\text{разбиение}(A, \emptyset)))$

6. Отбрасывание подмножества, не включающегося в базовое множество разбиения.

   $\forall_{Aabx}(\neg(a \subseteq A) \;\&\; \text{разбиение}(A, x) \rightarrow x \subseteq \{a;b\} \leftrightarrow x \subseteq \{;b\})$

   Преобразуемое включение является условием задачи на описание; x - неизвестная этой задачи; a, b, A - известны. Прием ускоряет перебор при поиске подсемейств, образующих разбиение A.

7. Разбиение пустого множества.

   разбиение$(\emptyset, \emptyset)$

   $\forall_{ab}(\neg(\emptyset \in \{;b\}) \;\&\; a \subseteq \{;b\} \rightarrow \text{разбиение}(\emptyset, a) \leftrightarrow a = \emptyset)$

Приемы символа "слойсемейства"

В заключение подраздела "семейства множеств" рассмотрим несколько приемов, связанных с понятием "слойсемейства". Это понятие часто встречается в тех задачах теории вероятности, где рассматривается вероятность появления заданного числа k событий заданного класса. Приводимые ниже приемы обязаны своим происхождением нескольким таким задачам. Еще раз напомним, что запись "слойсемейства(S A k)" обозначает множество всех элементов множества A, принадлежащих ровно k классам семейства множеств S.

1. Разложение по одному из множеств семейства.

   $\forall_{ABCi}(0 < i \rightarrow \text{слойсемейства}(A; B, C, i) = A \cap \text{слойсемейства}(B, C, i-1) \cup \text{слойсемейства}(B, C, i) \setminus A)$

   Преобразование применяется слева направо. Оно применяется в тех случаях, когда семейство задано явным перечислением элементов. После серии срабатываний приема слой семейства оказывается выражен через множества семейства с помощью объединения, пересечения и разности.

2. Явное выражение через объединение и пересечение семейств.

   $\forall_{ABin}(l(A) = n \,\&\, i \in \{0, \ldots, n\} \rightarrow \text{слойсемейства}(A, B, i) = \bigcup_{m, m \subseteq \{1,\ldots,n\}, \text{card} m = i}(\bigcap_{j, j \in m} A(j) \cap \bigcap_{j, j \in \{1,\ldots,n\} \setminus m}(B \setminus A(j))))$

   Прием применяется, если число n элементов семейства A известно и представлено в виде десятичной константы.

3. Семейство подмножеств прямого произведения.

   $\forall_{APQMNk}(M = \text{set}_i(P(i)) \rightarrow \text{слойсемейства}(\lambda_j(\bigcup_{i,P(i)}\{i\} \times A(i,j), Q(j)), M \times N, k) = \bigcup_{i,P(i)}(\{i\} \times \text{слойсемейства}(\lambda_j(A(i,j), Q(j)), N, k)))$

   Преобразование применяется слева направо. Первый антецедент обрабатывается вспомогательной задачей на доказательство, имеющей максимальный уровень 4.

4. Пустое семейство множеств.

   $\forall_a(\text{слойсемейства}(\text{пустоеслово}, a, 0) = a)$

   $\forall_{ab}(0 < b \rightarrow \text{слойсемейства}(\text{пустоеслово}, a, b) = \emptyset)$

5. Нулевой слой.

   $$\forall_{AnB}(l(a) = n \rightarrow \text{слойсемейства}(A, B, 0) = B \setminus \bigcup_{i=1}^{n} A(i))$$

   Наравление замены - слева направо. Выражение A имеет заголовок "набор" и задает семейство множеств явным перечислением. Указатель "развертка(фикс(0 2 2))" определяет развертку объединения семейства в обычное объединение.

6. Объединение всех слоев.

   $$\forall_{ABn}(l(A) = n \rightarrow \bigcup_{i=0}^{n} \text{слойсемейства}(A, B, i) = B)$$

$$\forall_{ABn}(l(A) = n \rightarrow \text{слойсемейства}(A, B, 0) \cup \bigcup_{i=1}^{n} \text{слойсемейства}(A, B, i) = B)$$

В первом приеме объединение семейства слоев идентифицируется непосредственно, во втором - выделено указателем "развертка" и идентифицируется с обычным объединением слоев. Член "слойсемейства$(A, B, 0)$" вынесен отдельно, чтобы определить выражения A, B до цикла идентификации остальных членов.

7. Усмотрение слоя семейства.

$\forall_{ABn}(l(A) = n \rightarrow \mathrm{set}_x(x \in B \ \& \ \mathrm{set}_y(y \in \{1, \ldots, n\} \ \& \ x \in A(y) \ \& \ A(y) - \mathrm{set}) = \emptyset) = \text{слойсемейства}(A, B, 0))$

$\forall_{ABn}(l(A) = n \rightarrow \mathrm{set}_x(x \in B \ \& \ \mathrm{card}(\mathrm{set}_y(y \in \{1, \ldots, n\} \ \& \ x \in A(y) \ \& \ A(y) - \mathrm{set})) = p) = \text{слойсемейства}(A, B, p))$

$\forall_{ABn}(l(A) = n \rightarrow \mathrm{set}_x(x \in B \ \& \ \mathrm{card}(\mathrm{set}_y(y \in \{1, \ldots, n\} \ \& \ \neg(x \in A(y)) \ \& \ A(y) - \mathrm{set})) = p) = \text{слойсемейства}(A, B, n - p))$

4.15 Приемы символа "класс"

Описатель "класс$(x_1 \ldots x_n P(x_1 \ldots x_n))$" задает множество всех наборов значений переменных $(x_1 \ldots x_n)$, удовлетворяющих условию $P(x_1 \ldots x_n)$. Если $n = 1$, то вместо одноэлементного набора (x_1) берется само значение x_1. Чтобы не допустить возникновения известных парадоксов теории множеств, проистекающих из неограниченного использования описателя, внесем в это соглашение следующее уточнение. Если существуют такие ранее определенные множества $M_1, \ldots, M_n$ что из истинности $P(x_1 \ldots x_n)$ вытекает принадлежность каждого значения x_i множеству M_i, то значением описателя действительно служит указанное выше множество наборов. Иначе - значением описателя является какое-то множество, о котором дополнительно ничего не известно. Приемы обычно предполагают выполненным первое предположение, рассматривая его как о.д.з. для описателя "класс".

Решение вспомогательной задачи на описание для условия на принадлежность классу

Если находящееся под описателем "класс" условие явно разрешить относительно переменных его связывающей приставки, то часто удается исключить этот описатель, перейдя к какому-либо элементарному выражению. Попытки такого разрешения предпринимаются несколькими различными приемами, предназначенными для задач различных типов. Четыре из этих приемов имеют одну и ту же теорему:

$\forall_f(\mathrm{set}_x(f(x)) = \mathrm{set}_x(f(x)))$

Начнем с приема, применяемого для описателей "класс", встречающихся в условии задачи на описание. Уровень срабатывания приема равен 2. Фильтры блокируют срабатывание приема в тех случаях, когда он может разрушить принятую в контексте стандартизацию. Например, прием нецелесообразно применять к описателям, под которыми расположены уравнения кривых. Заблокировано также применение приема для утверждений $f(x)$, имеющих заголовок "существует". Заменяющий терм приема получается из заменяемого только применением нормализатора к подвыражению $f(x)$. Роль этого нормализатора играет задача на описание, имеющая неизвестную x и цели "полный", "явное", "прямойответ", "упростить", "одз", "(класс x пустоеслово)". Последняя цель позволяет распознать задачу, решаемую для упрощения описателя "класс". Уровень обращения к задаче равен 7. Если описатель находится под символом "мощность" либо "конечное", то вводятся также цели "мощность" либо "плюсбеск". Эти цели используются несколькими приемами, целесообразность применения которых зависит от указанного внешнего контекста.

Аналогичные приемы созданы для случаев задач на преобразование и на исследование. Кроме того, в случае задачи на описание введен прием, имеющий теорему вида:

$\forall_f(\mathrm{set}_x(\exists_y(f(x,y))) = \mathrm{set}_x(\exists_y(f(x,y))))$

Здесь утверждение $f(x,y)$ разрешается относительно объединенного списка неизвестных x, y.

Расшифровка условия принадлежности классу

Несколько приемов выполняют расшифровку условия принадлежности элемента классу. Простейший из них имеет следующую теорему:

$\forall_{xf}(x \in \mathrm{set}_y(f(y)) \leftrightarrow f(x))$

Заметим, что переменная x не выделена указателем "кортежпеременных", и кроме данного приема введены еще несколько аналогичных - для принадлежности классу пары, тройки и четверки. Все эти преобразования выполняют общую стандартизацию. Однако, если имеется условие принадлежности классу наборов выражения, заголовок которого отличается от символа "набор", расшифровка его предпринимается лишь в специальных случаях. Приведем в качестве примера прием, основанный на следующей теореме:

$\forall_{fga}(a \in \mathrm{set}_{xyz}(x = f(y,z) \ \& \ g(y,z)) \leftrightarrow \exists_{yz}(g(y,z) \ \& \ a = (f(y,z), y, z)))$

Он применяется к условию задачи на описание, причем a должно содержать неизвестные. Прием заменяет условие принадлежности классу на параметрическое описание серии значений неизвестного выражения и сопровождает это описание комментариями "серия" и "фильтрсерии", обеспечивающими последующее явное разрешение данного описания относительно неизвестных задачи.

Переход к объединению множеств

Если конъюнктивным членом группы условий принадлежности классу является дизъюнкция, то класс заменяется на объединение классов:

$\forall_{fgh}(\mathrm{set}_x((f(x) \ \vee \ g(x)) \ \& \ h(x)) = \mathrm{set}_x(f(x) \ \& \ h(x)) \cup \mathrm{set}_x(g(x) \ \& \ h(x)))$

Объединяемые выражения предварительно обрабатываются нормализатором "нормкласс", имеющим двойник данного приема. Таким образом, фактически обработка "длинных" дизъюнкций происходит за один шаг. Если дизъюнкция не зависит от переменных связывающей приставки, прием блокируется.

Особо выделен подслучай дизъюнкции, имеющей явное указание элементов класса:

$\forall_{af}(\mathrm{set}_x(x = a \ \vee \ f(x)) = \{a\} \cup \mathrm{set}_x(f(x)))$

Подмножества целых чисел

Для переформулировки описателя "класс" в терминах простейших подмножеств целых чисел служат следующие приемы:

1. $\mathrm{set}_x(x - \text{целое}) = \mathrm{Z}$
2. $\mathrm{set}_x(x - \text{натуральное}) = \mathrm{N}$
3. $\forall_n(n - \text{натуральное} \to \mathrm{set}_i(i - \text{натуральное} \ \& \ i \leq n) = \{1, \ldots, n\})$
4. $\forall_{ab}(\mathrm{set}_x(x - \text{целое} \ \& \ a \leq x \ \& \ x \leq b) = \{-[-a], \ldots, [b]\})$
5. $\forall_{ab}(\mathrm{set}_x(x - \text{целое} \ \& \ a < x \ \& \ x \leq b) = \{[a] + 1, \ldots, [b]\})$
6. $\forall_{ab}(\mathrm{set}_x(x - \text{натуральное} \ \& \ a \leq x \ \& \ x \leq b) = \{-[-a], \ldots, [b]\})$
7. $\forall_m(\mathrm{set}_k(k - \text{натуральное} \ \& \ 0 \leq m - k) = \{1, \ldots, [m]\})$
8. $\forall_m(m - \text{целое} \to \mathrm{set}_k(k - \text{натуральное} \ \& \ k \leq m) = \{1, \ldots, m\})$
9. $\forall_m(m - \text{натуральное} \to \mathrm{set}_x(x - \text{натуральное} \ \& \ (m-1)/2 \leq x \ \& \ x \leq \ (m+1)/2) = (\{m/2\}\text{при } m - \mathrm{even}, \text{иначе } \{(m-1)/2, (m+1)/2\}))$

Неполнота и некоторая бессистемность приведенного списка объясняются тем, что он пополнялся от случая к случаю - по мере того, как в обучающих задачах возникала такая потребность.

Усмотрение пустого множества

$\mathrm{set}_x(\text{ложь}) = \emptyset$

Переход к числовым промежуткам

Следующие приемы усмотреть числовой промежуток, заданный с помощью описателя "класс".

1. $\mathrm{set}_a(a - \text{число}) = \mathrm{R}$
2. $\forall_b(\mathrm{set}_a(a - \text{число} \ \& \ a < b) = (-\infty, b))$

   Кроме этого, имеются еще три аналогичных приема - для случая нестрогого неравенства, а также для случая правого луча.

3. $\forall_{bc}(\mathrm{set}_a(a - \text{число} \ \& \ b < a \ \& \ a < c) = (b, c))$

   Еще три аналогичных приема введены для случаев, когда одно или два неравенства - нестрогие.

4. $\forall_a(\mathrm{set}_x(x - \text{число} \ \& \ |x| \leq a) = [-a, a])$

Усмотрение одноэлементного множества

1. $\forall_a(\mathrm{set}_b(b=a)=\{a\})$
2. $\forall_{ac}(\mathrm{set}_b(b=a\ \&\ c)=(\{a\}\text{при } c, \text{иначе } \emptyset))$

   Заметим, что здесь условие c не содержит переменной b; чтобы рассмотреть также случай, когда оно зависит от b, введен следующий прием.
3. $\forall_{af}(\mathrm{set}_x(x=a\ \&\ f(x))=(\{a\}\text{при } f(a), \text{иначе } \emptyset))$
4. $\forall_{abf}(\mathrm{set}_{xy}(x=a\ \&\ y=b\ \&\ f(x,y))=(\{(a,b)\}\text{при } f(a,b), \text{иначе } \emptyset))$
5. $\forall_{abc}(\mathrm{set}_{xyz}(x=a\ \&\ y=b\ \&\ z=c)=\{(a,b,c)\})$

   Аналогичный прием введен также для четырех переменных.

Исключение конъюнктивных членов, не зависящих от переменных связывающей приставки

$\forall_{af}(\mathrm{set}_x(a\ \&\ f(x))=(\mathrm{set}_x(f(x))\text{при } a, \text{иначе } \emptyset))$

Расшифровка включения в класс

$\forall_{AB}(B\subseteq \mathrm{set}_x(A(x))\leftrightarrow \forall_x(x\in B\to A(x)))$

Прием применяется к условию задачи на описание, причем в утверждение $A(x)$ должны входить квантор и неизвестная.

$\forall_{AB}(\mathrm{set}_x(A(x))\subseteq \mathrm{set}_y(B(y))\leftrightarrow \forall_x(A(x)\to B(x)))$

Включение должно содержать неизвестные задачи на описание.

Параметрическое описание класса

Параметрическое описание класса имеет вид $\mathrm{set}_x(\exists_y(A(y)\ \&\ x=f(y)))$. Здесь переменная или переменные y играют роль параметров описания, определяющих многообразие допустимых значений переменной (или переменных) x. Перечислим несколько приемов, относящихся к таким описаниям.

1. Ввод вспомогательного целочисленного параметра. Если первоначальное описание класса не являлось параметрическим, но содержало условие целочисленности значения некоторого варьируемого выражения, то вводится целочисленный параметр - значение данного выражения, и описание класса преобразуется к параметрическому виду. Имеется два таких приема:

   $\forall_{fg}(\mathrm{set}_x(f(x)\ \&\ g(x)-\text{целое})=\mathrm{set}_x(\exists_n(n-\text{целое}\ \&\ g(x)=n\ \&\ f(x))))$

   $\forall_{fgP}((f(x)=n\ \&\ g(x))=P\to \mathrm{set}_x(g(x)\ \&\ f(x)-\text{целое})=\mathrm{set}_x(\exists_n(n-\text{целое}\ \&\ P)))$

   В первом случае решается задача на описание; выражение "класс(...)" встречается в ее условии и содержит неизвестные. Переменная x входит в выражение $g(x)$ и отлична от него. Подкванторные утверждения в заменяющем параметрическом описании явно разрешаются относительно x с помощью вспомогательной задачи.

Во втором случае решается задача на преобразование, и выражение "класс(...)" служит ее условием. Антецедент присваивает переменной P результат явного разрешения утверждений $f(x) = n, g(x)$ относительно неизвестной x. Вспомогательная задача на описание, используемая при этом (см. нормализаторы, обслуживающие левую часть равенства в антецеденте), сопровождается дополнительной посылкой "целое(n)".

2. Устранение вырожденной параметризации.

   Если варьируемая переменная описателя "класс" определяется явным образом как набор значений параметров, то параметрическое описание исключается:

   $\forall_A(\mathrm{set}_x(\exists_{yz}(x = (y, z)\ \&\ A(y, z))) = \mathrm{set}_{yz}(A(y, z)))$

3. Преобразование параметрического описания при вычитании множеств.

   Если из множества, заданного параметрическим описанием, вычитается некоторое другое множество, то предпринимается попытка упростить разность, преобразовав ее тоже к виду параметрического описания. Теорема приема имеет вид:

   $\forall_{fgaP}(P = (g(y)\ \&\ \neg(f(y) \in a)) \to \mathrm{set}_x(\exists_y(x = f(y)\ \&\ g(y))) \setminus a = \mathrm{set}_x(\exists_y(x = f(y)\ \&\ P)))$

   Прием применяется на уровне 5; разность должна входить в условие задачи на преобразование. Первый антецедент присваивает переменной P результат упрощения конъюнкции $g(y)\ \&\ \neg(f(y) \in a)$ с помощью вспомогательной задачи на преобразование. Замена предпримается лишь в случае, если данный результат короче исходной конъюнкции.

   Если параметрическое описание вычитается из множества, заданного посредством описателя "класс", то первое заносится во второе как серия исключаемых точек:

   $\forall_{fghA}(\mathrm{set}_x(f(x)) \setminus (\mathrm{set}_y(\exists_z(y = g(z)\ \&\ h(z))) \cup A) = \mathrm{set}_x(f(x)\ \&\ \forall_z(h(z) \to \neg(x = g(z)))) \setminus A)$

   Уровень срабатывания приема равен 6. Вспомогательные задачи здесь не используются.

4. Переход от условия принадлежности перечню к дизъюнкции равенств.

   Если множество определено посредством обобщенного параметрического описания, в котором условие на его текущий элемент имеет не вид равенства, выражающего этот элемент через параметры, а вид принадлежности списку объектов, заданных через параметры, то выражение преобразуется к виду объединения множеств, заданных обычными параметрическими описаниями:

   $\forall_{fgh}(\mathrm{set}_x(\exists_y(x \in \{f(y); g(y)\}\ \&\ h(y))) = \mathrm{set}_x(\exists_y(x = f(y)\ \&\ h(y))) \cup \mathrm{set}_x(\exists_y(x \in \{; g(y)\}\ \&\ h(y))))$

5. Занесение под квантор существования внешних условий.

   Если под описателем "класс" расположена конъюнкция квантора существования, выражающего элемент множества через параметры, а также дополнительного условия на данный элемент, то последнее заносится под квантор существования для перехода к виду "стандартного" параметрического описания:

$$\forall_{fPQ}(\mathrm{set}_x(\exists_y(x = f(y)\ \&\ P(y))\ \&\ Q(x)) = \mathrm{set}_x(\exists_y(x = f(y)\ \&\ P(y)\ \&\ Q(f(y)))))$$

Так как данная стандартизация оказалась нужной лишь в комбинаторике, необходимым условием срабатывания приема является наличие в текущем терме символа "мощность".

6. Представление подмножества объединения в виде объединения подмножеств.

   При определении мощности множества подмножеств A некоторого множества M, удовлетворяющих определенным условиям, иногда бывает полезно разбивать M на две непересекающиеся части и составлять A из подмножеств этих частей. Здесь введены следующие два приема:

   $$\forall_{BCP}(\text{непересек}(B,C) \rightarrow \mathrm{set}_A(A \subseteq B \cup C\ \&\ P(A)\ \&\ A-\mathrm{set}) = \mathrm{set}_A(\exists_{DE}(D \subseteq B\ \&\ E \subseteq C\ \&\ A = D \cup E\ \&\ D-\mathrm{set}\ \&\ E-\mathrm{set}\ \&\ P(D \cup E))))$$

   $$\forall_{BCP}(B \subseteq C \rightarrow \mathrm{card}(\mathrm{set}_{xy}(x \subseteq C\ \&\ P(x,y)\ \&\ x-\mathrm{set})) = \mathrm{card}(\mathrm{set}_{zvy}(z \subseteq B\ \&\ v \subseteq C \setminus B\ \&\ z-\mathrm{set}\ \&\ v-\mathrm{set}\ \&\ P(z \cup v, y))))$$

   В первом случае предполагается, что описатель "класс" расположен непосредственно под символом "мощность". В первом случае под описателем должны найтись выражения $A \cap B$, $A \cap C$; во втором - выражение $x \cap B$. Переменная y во втором приеме выделена указателем "кортежпеременных".

Переход к перечню пар

Если описатель "класс" задает множество пар, причем возможные значения одного элемента пары перечислены в конечном списке, а другой элемент однозначно определяется по первому, то выражение преобразуется к виду конечного списка пар:

$$\forall_{fab}(\mathrm{set}_{xy}(x = f(y)\ \&\ y \in \{a;b\}) = \{(f(a),a)\} \cup \mathrm{set}_{xy}(x = f(y)\ \&\ y \in \{;b\}))$$

Описатель в заменяющем терме обрабатывается нормализатором "нормкласс", который имеет аналогичный прием и доводит до конца обработку всех элементов списка b. Нормализатор "нормобъединение", примененный к заменяющему терму, сворачивает одноэлементые множества, составленные из пар, в конечный список этих пар. Чтобы привести к стандартному виду выражение $f(a)$ в каждой паре, эти пары обрабатываются вспомогательными задачами на упрощение. Имеется двойственный прием для случая, когда конечным списком перечисляются первые элементы пар.

Усмотрение прямого произведения

$$\forall_{fg}(\mathrm{set}_{xy}(f(x)\ \&\ g(y)) = \mathrm{set}_x(f(x)) \times \mathrm{set}_y(g(y)))$$

Несмотря на то, что данное преобразование выглядит как вполне естественная общая стандартизация, имеется множество специальных случаев, где его нужно блокировать. Они перечислены в фильтрах приема. Например, преобразование бывает невыгодным при работе с уравнениями кривых в аналитической геометрии; при определении уравнений границ для области интегрирования, и т.п. Оно нарушает стандартизацию, необходимую для срабатывания приемов, применяемых в этих разделах.

Ориентация равенства с классом

В решателе принята автоматическая ориентация частей равенства, выполняемая описанным ранее общелогическим приемом. На основе этой ориентации далее выполняется стандартизация обозначений: повсюду, где данное равенство присутствует в контексте, вхождения его левой части заменяются на правую часть. Однако, автоматическая ориентация равенств часто не учитывает специфики ситуации и требует особой коррекций. Она выполняется различными приемами, переставляющими части равенства и сопровождающими его комментарием "ориентацияравенства". Этот комментарий блокирует автоматическую ориентацию.

В случае описателя "класс" часто бывает выгодно ввести для него специальное обозначение - некоторую вспомогательную переменную a, сохранить в посылках равенство, определяющую данную переменную, после чего в прочих местах заменить описатель на его обозначение a. Однако, автоматическая ориентация будет действовать обратным образом - подставит вместо всех вхождений обозначения a явное выражение с описателем. Для блокировки этого созданы два приема, имеющие следующую теорему:

$\forall_{af}(a = \mathrm{set}_x(f(x)) \leftrightarrow \mathrm{set}_x(f(x)) = a)$

В обоих случаях равенство представляет собой посылку задачи, причем уровень срабатывания равен 0. Приемы отличаются друг от друга лишь фильтрами, уточняющими контекст срабатывания. Строго говоря, части равенства переставляются не всегда - они изначально уже могли быть ориентированы нужным образом. Однако, отсутствие комментария "ориентацияравенства" приводит к срабатыванию приема и в таком случае. Само равенство тогда не изменяется, но комментарий вводится и блокирует последующую автоматическую перестановку, которая испортила бы требуемую стандартизацию.

Подстановка явно заданного значения переменной

Если значение одной из переменных связывающей приставки описателя "класс" явно задано равенством, то оно подставляется вместо всех остальных вхождений переменной под описателем:

$\forall_{af}(\mathrm{set}_{xy}(x = a \ \&\ f(x, y)) = \mathrm{set}_{xy}(x = a \ \&\ f(a, y)))$

Пересечение двух классов

Пересечение двух классов преобразуется к виду единственного описателя "класс":

$\forall_{AB}(\mathrm{set}_x(A(x)) \cap \mathrm{set}_y(B(y)) = \mathrm{set}_x(A(x) \ \&\ B(x)))$

Требуется, чтобы связывающие приставки x, y имели одинаковую длину и чтобы утверждения $A(x), B(y)$ не начинались с квантора существования.

Разбор случаев по известному подутверждению описателя "класс"

Если в условии задачи на описание встречается выражение $\mathrm{set}_x((A \ \&\ f(x)) \ \vee\ g(x))$, где утверждение A не содержит неизвестных и не зависит от связывающей приставки x, то предпринимается разбор случаев по истинности A. Теорема этого приема имеет вид $\forall_A(A \ \vee\ \neg A)$; заголовок приема - "вывод". Чтобы задать инициализацию приема при усмотрении указанного выше выражения, используется указатель "контрольвывода(класс(x23 или(и(x26 значение(x6 x23))значение(x7 x23))))". Указатель "примечание(разборслучаев)" форсирует разбор случаев по выведенной дизъюнкции на малых уровнях сканирования. Так как уровень срабатывания приема равен 0, то нужно предпринять меры против зацикливания: разбор случаев не будет успевать за повторными срабатываниями данного приема. Этой цели служат фильтры "не(Входит(или списокусловий))", "не(контекст(новоеусловие(x1) вхождениетерма(корень x1)))", "не(контекст(новоеусловие(x1)вид(x1 не(x2))вхождениетерма(корень x2)))". Рекомендуем в качестве упражнения найти те задачи, где данный прием используется, и проанализировать возможность увеличения уровня его срабатывания.

Специальные приемы

Кроме перечисленных выше приемов описателя "класс", имеющих сравнительно общий характер, в разделе собраны несколько десятков приемов более специальной ориентации. Они возникли при рассмотрении различных задач теории вероятностей ("Класс объектов, принадлежащих не более чем одному элементу заданного семейства"), математического анализа ("Разрешение числового равенства под описателем относительно одной из переменных"), алгебры логики ("Действия с классами наборов"), аналитической геометрии ("Общая стандартизация уравнения кривой либо поверхности"), и т.п. Представляем читателю самостоятельно вернуться к этим приемам в последующих разделах книги.

Нормализатор "нормкласс"

Для общей стандартизации выражений, имеющих заголовок "класс", служит нормализатор "нормкласс". В нем собраны дубликаты ряда приведенных выше простейших приемов, исключающих описатель либо выражающих его через более простые описатели. Обращение к вспомогательной задаче для явного разрешения условия принадлежности классу здесь не используется.

4.16 Упражнения

Приведем несколько несложных упражнений на обучение решателя приемам, относящимся к рассмотренным выше дополнительным понятиям алгебры множеств.

1. Ввести приемы, необходимые для решения относительно x, y уравнения $x \times y = \{(1,1),(1,2),(2,1),(2,2)\}$.
2. Ввести новый тип целевой установки задач на преобразование: "представить теоретико-множественное выражение в виде прямого произведения". Научить систему решать задачи на представление в виде прямого произведения перечня пар - например, $\{(1,1),(1,2),(2,1),(2,2)\}$.
3. Ввести приемы, необходимые для разрешения относительно неизвестной x условия $\forall_y (y \in a \;\&\; x \in y \to \forall_z (z \in b \to x \in z))$.
4. Ввести задачу на нахождение всех разбиений множества $\{1,2,3\}$. Создать приемы, необходимые для ее решения.

5. Ввести приемы, необходимые для решения уравнения:

   слойсемейства$((\{1,2\},\{1,3\},\{2,4\}),\{1,2,3,4\},x)=\{1,2\}$.

6. Ввести прием, заменяющий условие $(a_1,\ldots,a_n)\in B_1\times\ldots B_n$ на условие $a_1\in B_1$ & ... & $a_n\in B_n$ для произвольного натурального $n\geq 2$.

4.16.1 Указания

1. Вводим уравнение и убеждаемся, что система его не решает. В разделе "Прямоепроизведение" базы приемов находим приемы, связанные с решением уравнений, одна из частей которых имеет вид прямого произведения. Они обращаются к оператору "видпрямоепроизведение", предпринимающему попытку представить в виде прямого произведения и другую часть. Включаем трассировку и выходим на момент обращения к оператору "видпрямоепроизведение", Таким образом убеждаемся, что система в данной задаче действительно к нему обращается. Далее остается лишь пополнить оператор "видпрямоепроизведение", чтобы он мог разложить на множители выражение $\{(1,1),(1,2),(2,1),(2,2)\}$. Заметим, что для разложения в прямое произведение множества пар имеется очень простой способ - нужно найти первую и вторую проекции, перемножить их и сравнить с исходным множеством. Реализуем этот способ на ГЕНОЛОГе. Пусть рассматривается множество пар A. Во-первых, нам нужно найти обе его проекции. Введем для этого антецеденты $a=$ Проекция$(A,1)$, $b=$ Проекция$(A,2)$. Если A задано конечным списком, то обращение к нормализатору "нормПроекция" позволит вычислить a и b. Так как включение $A\subseteq a\times b$ заведомо выполнено, достаточно проверить обратное включение, введя в теорему приема антецедент $a\times b\subseteq A$. Применим к левой части включения нормализатор "нормпрямоепроизведение(замечание(нормперечень))", который преобразует произведение двух конечных списков в список пар. Сам антецедент выделим указателем "блокпроверок". Тогда он будет обрабатываться проверочным оператором "усмсодержится". Наконец, в качестве консеквента теоремы приема возьмем равенство $A=a\times b$. Окончательно, теорема приема будет иметь вид:

   $\forall_{Aab}(a=$ Проекция$(A,1)$ & $b=$ Проекция$(A,2)$ & $A\subseteq a\times b\rightarrow A=a\times b)$.

   Заголовок приема - "замена(второйтерм видпрямоепроизведение)". Введя и откомпилировав прием, проверям, что уравнение системой решается. Созданный прием сохраняем, так как он понадобится уже в следующей задаче.

2. Прежде всего, нужно выбрать логический символ, который будет кодировать указанную цель задачи на преобразование. Например, пусть это будет символ "прямоепроизведение". Проверяем по связанной с символом справочной информации, что он ранее не был использован как элемент целевой установки. Теперь нам нужно пополнить меню целевых установок системы. Входим в какую-либо из задач задачника и вызываем данное меню для редактирования (клавиша Shift-ц). Входим в подраздел "преобразовать выражение", вводим новый концевой пункт "разложить в прямое произведение", нажимаем "курсор вправо" и "Enter", после чего вводим текстовым редактором кодовый символ "прямоепроизведение". При выборе данного пункта оглавления система будет обращаться к справочнику "смцель", который должен организовать диалог ввода параметров целевой установки. Чтобы получить образец приемов такого справочника,

выбираем какой-либо другой концевой пункт оглавления и находим его кодовый символ. Например, пусть это будут пункт "разложить перестановку в произведение независимых циклов" и его кодовый символ "циклперест". Находим прием справочника "смцель" для символа "циклперест" и копируем его в программу символа "прямоепроизведение". В этой копии нужно заменить ссылку на текст, который будет выдаваться на экран при выборе целевой установки. Такой текст (т.е. "представить теоретико-множественное выражение в виде прямого произведения") нужно создать во 2-м инф.блоке. Входим в подраздел "ресурсы и установки" главного меню; нажимаем "и" и вводим номер "2" нужного информационного блока. После нажатия "Enter" оказываемся в корневом каталоге блока. Мы должны выбрать какой-либо логический символ S, такой, что из корневого каталога нет цепочки переходов по символам S, "слово". Пытаемся угадать S. Например, пусть сначала будет $S =$ "прямоепроизведение". Нажимаем "курсор вниз" и вводим данный символ. Оказываемся в некотором ранее созданном указателе - списке. Перехода по символу "слово" из него нет, так что выбранное S нам подходит. Нажимаем "Insert" и вводим символ "слово". Сверху появляется меню, в котором выбираем пункт "Ч.-б. текст". Наконец, вводим текстовым редактором шаблон "Представить теоретико-множественное выражение в виде прямого произведения". Набор завершаем нажатием "Enter", и далее нажимаем "End" для возвращения в главное меню.

Роль ссылки на введенный текстовый шаблон играет символ "прямоепроизведение". Возвращаемся в скопированную программу справочника "смцель" и заменяем в операторе "смцель(набор(Выч)x1 x2 x3)" символ "Выч", который представлял собой ссылку на текстовый шаблон исходной программы, на символ "прямоепроизведение". В новой программе нужно также скорректировать список целей вводимой задачи - вместо старого комплекта "упростить, одз, циклперест" нужен комплект "упростить, одз, прямоепроизведение". На этом коррекция справочника "смцель" завершена. Можно попробовать воспользоваться оглавлением целей и ввести новую задачу. Например, пусть это будет задача на разложение выражения $\{(1,1),(1,2),(2,1),(2,2)\}$. Однако, если после набора задачи выйти в оглавление задачника, а затем снова вернуться в просмотр задачи, то окажется, что текстовый шаблон изменился - теперь он имеет вид "Упростить выражение". Нажатием "Ctr-ц" можно убедиться, что целевая установка введена верно, и причину рассогласования следует искать в программах интерфейса задачника. Возвращаемся в главное меню, входим в оглавление программ и выбираем раздел "Интерфес просмотра и редактирования задач" - "Процедура ЭЛЕМЕНТЫЗАДАЧИ" - "Стандартные комплекты целей задач на преобразование". В этой точке находится программа, которая усматривает один из стандартных комплектов целей задачи и формирует для его прорисовки соответствующую текст-формульную строку. Чтобы долго не искать нужный нам случай, будем действовать по аналогии. Нажимаем F4 и вводим символ "циклперест" - элемент целевой установки, который мы выше использовали как образец. Появляется фрагмент программы "или(цель(x1 разложитьнамножители)цель(x1 циклперест)...)". Добавляем в эту дизъюнкцию оператор "цель(x1 прямоепроизведение)". Чтобы определить кодовый символ x7 нужного нам текстового шаблона, разветвляем последний пункт цепочки вложенных условных выражений в правой части присвоения: заменяем символ "видеоключ" на терм "вариант(цель(x1 прямоепроизведение)прямоепроизведение видеоключ)". Теперь можно вернуться в просмотр нашей задачи и убедиться, что текстовый шаблон прорисован верно.

Чтобы при решении задачи с целью "прямоепроизведение" произошло обращение к нормализатору "видпрямоепроизведение", нужно ввести специальный прием. К сожалению, здесь нет никакого конкретного логического символа, появление которого в преобразуемом выражении инициировало бы попытку применения данного приема. Поэтому возможны два пути: либо создать на ЛОСе общий прием задач типа "преобразовать", который будет срабатывать при обнаружении цели "прямоепроизведение", либо выделить несколько типичных символов - заголовков выражений, к которым будет применяться прием, и создать для них различные приемы на ГЕНОЛОГе.

Реализуем второй способ. В нашем случае заголовком преобразуемого выражения служит символ "перечень". Поэтому берем следующую теорему приема: $\forall_{ab}(a = \{;b\} \rightarrow \{;b\} = a)$. Антецедент выделяем указателем "идентификатор(1)", к правой его части применяем нормализатор "видпрямоепроизведение". Вводим фильтр "заголовок(x1 прямоепроизведение)", контролирующий успех попытки разложения.

Согласно предыдущему упражнению, система теперь должна решить введенную задачу. Однако, после запуска процесса решения оказывается, что она зациклилась. Это обнаруживаем по затянувшейся паузе: нажимаем "Break" и видим, что после получения прямого произведения $\{1,2\} \times \{1,2\}$ применяется прием "элиминация прямого произведения перечней", выполняющий обратное преобразование. Чтобы данный прием учитывал нашу новую целевую установку, вводим в него дополнительный фильтр: "или(не(тип(преобразовать)) не(цель(прямоепроизведение)) посылка не(корень))". Снова запускаем процесс решения, и получаем ответ.

3. Вводим условие задачи и убеждаемся, что система ее не решает. Чтобы определить, на каком этапе возникают затруднения, начинаем трассировку. Система преобразует исходное утверждение к виду дизъюнкции $\forall_y(y - \mathrm{set}\ \&\ y \in a \rightarrow \neg(x \in y)) \vee \forall_z(z \in b \rightarrow (z - \mathrm{set}\ \&\ x \in z))$. Утверждение $z - \mathrm{set}$ в консеквенте второго дизъюнктивного члена введено системой для сопровождения по о.д.з. По этой причине преобразование второй кванторной импликации в конъюнкцию импликаций блокируется. На следующем шаге срабатывает прием, преобразующий первую кванторную импликацию к виду отрицания принадлежности объединению семейства множеств:

$$\neg(x \in \bigcup_{y,y-set,y\in a} y).$$

Следующий шаг - разбор случаев по дизъюнкции. Рассмотрение первого подслучая сразу приводит к получению частичного ответа. Однако, при рассмотрении второго подслучая система не может перейти к условию принадлежности неизвестной пересечению семейства. После нескольких безуспешных попыток применить прочие приемы здесь выдается отказ. Таким образом, наша задача проясняется: нужно создать прием, переформулирующий кванторную импликацию $\forall_z(z \in b \rightarrow (z - \mathrm{set}\ \&\ x \in z))$ в терминах принадлежности пересечению семейства. Прежде всего, попробуем найти в базе приемов близкие приемы. Переходим в раздел "Теория множеств" - "Семействамножеств" - "Перечесениевсех" - "Кванторная свертка в условие принадлежности пересечению семейства множеств". Оказывается, что там уже есть прием, выполняющий аналогичное преобразование. Он применяется к утверждениям вида $\forall_x(f(x) \rightarrow a \in g(x))$. Однако, применение его требует усмотрения непустоты множества таких x, что истинно $f(x)$. В нашем случае непустота не гарантирована, и требуется более общий прием. Чтобы учесть возможность появления в консеквенте заменяемой импликации дополнительных условий, нужных для сопровождения по о.д.з., будем считать, что она имеет вид $\forall_x(A(x) \rightarrow a \in f(x)\ \&\ B(x))$. Тогда получаем следующую теорему приема:

$\forall_{ABfa}(\forall_x(A(x) \rightarrow a \in f(x)\ \&\ B(x)) \leftrightarrow \forall_x(A(x) \rightarrow B(x))\ \&\ (\neg(\exists_x(A(x))) \vee \exists_x(A(x))\ \&\ a \in \bigcap_{x,A(x)} f(x)))$.

Утверждение $\exists_x(A(x))$ добавлено к условию принадлежности пересечению семейства как сопровождающее по о.д.з.

В фильтрах приема указываем, что выражение, идентифицируемое с a, содержит неизвестные, а прочие подвыражения - не содержат. Переменные A, B, f выделяем указателем "отображение", переменную x - указателем "кортежпеременных". Чтобы пополнить список нормализаторов приема, используем клавишу "н". Будут последовательно появляться предложения, которые можно выбирать либо отвергать нажатием клавиш Insert, Delete. Важно при этом следить за левой частью меню, размещенного в верхней части экрана. Если там записано "Новый указатель", то речь идет о добавлении к описанию приема каких-либо элементов. Если же появляются слова "Старый указатель", то процедура предлагает убрать или сохранить те элементы приема, которые не могут быть воспроизведены текущей версией генератора приемов. Обычно такие предложения отвергаются, и лучше сразу нажать Esc для обрыва последовательности предложений. Ранее отобранные либо удаленные элементы будут учтены в текущей просматриваемой версии приема, но пока не зарегистрированы в файлах. Чтобы их закрепить, нужно нажать, например, F3 (с перекомпиляцией) либо F4 (без перекомпиляции).

После того, как прием создан, снова запускаем процесс решения и убеждаемся, что ответ получен.

4. Задача на нахождение разбиений имеет единственное условие "разбиение($\{1, 2, 3\}, x$)" и неизвестную x. Чтобы перечислить разбиения, будем использовать рекурсию с отбрасыванием первого элемента перечня. Теорема приема, определяющая данную рекурсию, имеет следующий вид:

   $\forall_{abA}$(разбиение($\{;b\}, y$) $= A(y) \rightarrow$ разбиение($\{a;b\}, x$) $\leftrightarrow \exists_y(A(y)\ \&\ (x = \{\{a\}\} \cup y\ \vee\ \exists_z(z \in y\ \&\ x = (y \setminus \{z\}) \cup \{z \cup \{a\}\}))))$.

   В исходном условии "разбиение($\{a;b\}, x$)" отбрасывается первый элемент списка a и решается вспомогательная задача с неизвестной y, имеющая условие "разбиение($\{;b\}, y$)". Обращение к ее решению происходит при обработке антецедента теоремы, выделенного указателем "идентификатор". Левая часть антецедента помечена нормализатором "задача(4 тип(описать) полный явное прямойответ упростить цель(неизвестная(x24)))"; ответ задачи присваивается функциональной переменной $A(y)$. Заменяющий терм строится на основе утверждения $A(y)$, означающего, что y - некоторое разбиение подсписка $\{;b\}$. В нем рассматриваются два случая: либо x получено добавлением к y нового одноэлементного класса a, либо получено из y добавлением элемента a к одному из старых классов. Последнее выражено как результат исключения из y класса z и добавления класса $z \cup \{a\}$. Прием имеет фильтры "урвень(2)", "условие", "тип(описать)", "не(известно(x23))" и указатели "идентификатор(1)", "отображение(x26)", "новаяпеременная(x24)".

После создания приема и проверки того, что задача решена, рекомендуем рассмотреть задачу на перечисление всех разбиений множества $\{1, 2, 3, 4\}$. Хотя ответ и будет получен, он потребует дополнительных упрощений: система не сумеет вычитать друг из друга конечные списки, составленные из конечных списков. Создайте прием проверочного оператора "усмнепринадлежит", который позволил бы упростить данный ответ. Аналогичную работу проделайте для пятиэлементного множества.

5. Чтобы определить значение неизвестной x, достаточно найти, скольким множествам указанного семейства принадлежит какой-либо элемент результирующего множества, например, 1. Поэтому вводим следующую теорему приема:

   $\forall_{abcdxmn}(\text{слойсемейства}(\lambda_i(a(i), i \in \{1, \ldots, n\}), b, x) = \{c; d\}\ \&\ m = \sum_{i=1}^{n}(1 \text{ при } c \in a(i),\ \text{иначе } 0) \to x = m)$

   Прием будет определять дополнительное условие задачи $x = m$, поэтому заголовок его - "выводусловия". Фильтры - "уровень(3)", "тип(описать)", "условие", "не(известно(x23))", "известно(фикс(1 1 1))", "известно(x2)", "известно(x3)". Первый антецедент идентифицируется с рассматриваемым уравнением. Подтерм $\lambda_i(a(i), i \in \{1, \ldots, n\})$ выделен указателем "развертка", т.е. семейство множеств должно быть явно задано термом "набор(...)". Функциональная переменная a выделена указателем "отображение". При идентификации находится набор термов $a(1), \ldots, a(n)$, используемый компилятором для построения других выделенных указателем "развертка" термов с переменной a. Второй антецедент выделен указателем "идентификатор", причем входящая в него конечная сумма выделена указателем "развертка". Она обрабатывается нормализатором "нормплюс"; подвыражение "вариант(...)" обрабатывается нормализатором "нормвариант"; подутверждение $c \in a(i)$ - обрабатывается нормализатором "нормусм". Эти нормализаторы обеспечивают в стандартных случаях непосредственное вычисление количества элементов семейства a, которым принадлежит c. Компилируем прием; вводим условие задачи и убеждаемся, что она системой решается. Однако, если попробовать провести трассировку по шагам решения, то выяснится, что после нахождения условия $x = 2$ проверка равенства для слоя семейства на экран не выводится. Рекомендуется использовать режим трассировки "Ручной выбор входа в подпроцесс" для просмотра того, как будет выполняться проверка. Она оказывается чрезмерно сложной, и попытка добиться существенного ее упрощения оставляется читателю в качестве самостоятельного дополнительного упражнения.

6. Теорема приема имеет вид:

$$\forall_{aBn}(\lambda_i(a(i), i \in \{1, \ldots, n\}) \in \prod_{i=1}^{n} B(i) \leftrightarrow \forall_i(i \in \{1, \ldots, n\} \to a(i) \in B(i))).$$

   Знак произведения соответствует символу "Прямоепроизведение" во внутренней (текстовой) записи данной теоремы. Его можно вводить нажатием клавиши "Ctr-$*$". Указатель "развертка(фикс(0 1 1)фикс(0 1 2)фикс(0 2))" определяет идентификацию выражения $\lambda_i(a(i), i \in \{1, \ldots, n\})$ с конечным набором, идентфикацию выражения $\prod_{i=1}^{n} B(i)$ с прямым произведением n множеств, а также построение терма $\forall_i(i \in \{1, \ldots, n\} \rightarrow a(i) \in B(i))$ в виде конъюнкции принадлежностей. Чтобы предотвратить вырожденный случай одноэлементного набора, вводим фильтр "меньше(1 x14)".

Глава 5

Приемы, связанные с простейшими свойствами функций

Перейдем к рассмотрению общих приемов, связанных с функциями. Они собраны в разделе базы приемов "Теория множеств" - "Функции". Напомним используемые в решателе обозначения основных понятий, относящихся к этому разделу. Утверждение "функция(f)" означает, что f есть функция. Как и множества, функции представляют собой один из базисных типов объектов. В частности, никакая функция не является объектом другого базисного типа (множеством, числом, и т.п.). Значение функции f в точке a обозначается "значение(f a)", причем формульный редактор прорисовывает это выражение как $f(a)$. Область определения функции f обозначается "область(f)". Она прорисовывается в виде $\mathrm{Dom}(f)$. Множество значений функции f на ее области определения обозначается "значения(f)" и прорисовывается в виде $\mathrm{Val}(f)$. Как обычно, считаем две функции равными, если они имеют одну и ту же область определения, а значения их на любом элементе из данной области равны.

Множество значений функции f на элементах множества A обозначаем "образ(f A)". Для прообраза введены два различных обозначения. Множество всех элементов из области определения функции f, на которых она принимает значение a, обозначено "слой(f a)". Выражение "прообраз(f A)" задает множество всех элементов области определения функции f, на которых ее значение принадлежит множеству A. Утверждение "взаимнооднозначно(f)" означает, что значения функции f на различных элементах ее области определения различны. Наконец, утверждение "Отображение(f A B)" означает, что f есть функция, область определения которой равна A, а множество значений содержится в B. Прочие обозначения используются сравнительно редко, и мы рассмотрим их в нижеследующих подразделах. Раздел общих приемов для функций пополнялся от случая к случаю - по мере того, как в этом возникала необходимость при рассмотрении других разделов. Поэтому, как правило, количество связанных с понятиями раздела приемов очень невелико, причем сами приемы достаточно простые, а реализация их на ГЕНОЛОГе аналогична рассмотренным ранее случаям. Впрочем, это предоставляет читателю хорошую возможность для самостоятельных упражнений по продолжению обучения решателя.

5.1 Приемы символа "значение"

Ориентация равенства, задающего значение функции

Если посылка задачи представляет собой равенство выражений $A(t)$, B, где A, B - переменные, то эти выражения переставляются таким образом, чтобы $A(t)$ находилось слева. Исключение составляет лишь случай, когда B - обозначение геометрической точки. Теорема приема имеет вид $\forall_{fin}(f(i) = n \leftrightarrow f(i) = n)$. Равенство идентифицируется без учета порядка операндов, а заменяющий терм переупорядочивает их нужным образом. Прием сопровождает результирующую посылку комментарием "ориентацияравенства", блокирующим повторные срабатывания данного приема, а также обратные перестановки, которые могли бы быть выполнены рассмотренным ранее общелогическим приемом ориентации равенства.

Определение значения функции, заданной описателем "отображение"

Напомним, что для задания функции может быть использован терм вида "отображение($x_1 \ldots x_n\ t\ A$)", где t - выражение, A - утверждение. Функция считается определенной на множестве всех наборов значений переменных $x_1, \ldots, x_n$, для которых истинно утверждение A, причем значение ее на таком наборе равно значению выражения t. Формульный редактор прорисовывает данное обозначение в виде $\lambda_{x_1 \ldots x_n}(t, A)$. Чтобы можно было в теореме приема указывать термы, полученные из t, A подстановкой каких-либо новых выражений вместо переменных $x_1, \ldots, x_n$, обычно используется запись $\lambda_{x_1 \ldots x_n}(t(x_1 \ldots x_n), A(x_1 \ldots x_n))$. В случае $n = 1$ вместо наборов длины 1 берутся сами значения переменной x_1, удовлетворяющие условию A.

Если функция f задана указанным образом с помощью описателя "отображение", то выражение "значение($f\ a$)" можно заменить на явное выражение, извлекаемое из данного описателя. Имеется несколько приемов, выполняющих такую замену. Прежде всего, созданы два приема с одной и той же теоремой $\forall_{fuva}(f = \lambda_x(u(x), v(x)) \rightarrow f(a) = u(a))$. Заменяемое вхождение находится в условии задачи; антецедент идентифицируется с некоторым утверждением из контекста этого вхождения. Приемы частично заблокированы для случаев, когда $f(a)$ - непосредственный операнд другого описателя "отображение". Если выражение a содержит переменные, то применяется прием, у которого заменяющий терм обрабатывается вспомогательной задачей на упрощение, иначе - прием, у которого этот терм обрабатывается лишь процедурой "норм".

Еще один прием, аналогичный указанным, создан для работы с посылками задач на преобразование. Его теорема имеет вид $\forall_{Atpf}(f = \lambda_z(t(z), A(z))\ \&\ A(p) \rightarrow f(p) = t(p))$. Первый антецедент идентифицируется с посылкой; второй - обрабатывается вспомогательной задачей на доказательство. Заменяемый терм $f(p)$ расположен в равенстве, являющемся посылкой задачи на преобразование.

Для упрощения выражений типа "значение(отображение(...)...)" создан прием с теоремой: $\forall_{uva}(v(a) \rightarrow \lambda_x(u(x), v(x))(a) = u(a))$. Антецедент выделен указателем "следствие" и обрабатывается с помощью вспомогательной задачи на доказательство. Заменяющий терм упрощается с помощью задачи на преобразование.

Если отображение представлено как набор, то для определения его значений служит прием: $\forall_{bnj}(a = \lambda_i(b(i), i \in \{1, \ldots, n\})\ \&\ j \in \{1, \ldots, n\} \rightarrow a(j) = b(j))$. Чтобы терм $\lambda_i(b(i), i \in \{1, \ldots, n\})$ был идентифицирован с набором, введен указатель "развертка(фикс(1 2))". Второй антецедент обрабатывается проверочным оператором. Прием применяется без каких-либо ограничений.

Аналогичный прием имеется для отображений, представленных в виде матриц: $\forall_{abnmpq}(a = \lambda_{ij}(b(i,j), i \in \{1,\dots,n\}$ & $j \in \{1,\dots,m\})$ & $p \in \{1,\dots,n\}$ & $q \in \{1,\dots,m\} \to a(p,q) = b(p,q))$. Идентификацию отображения с матрицей, заданной в виде терма "строки(...)", определяет указатель "матрица(фикс(1 2))".

Усмотрение равенства функций

Если области определения двух функций равны, то кванторная импликация, выражающая равенство значений этих функций, заменяется на условие равенства функций:

$\forall_{fgM}(\mathrm{Dom}(f) = M$ & $\mathrm{Dom}(g) = M \to \forall_i(i \in M \to f(i) = g(i) \leftrightarrow f = g))$.

Антецеденты выделены указателями "идентификатор", а их левые части сопровождены нормализаторами "нормобласть". Это обеспечивает проверку совпадения областей определения.

Разрешение относительно неизвестной

Равенства $f(x) = a$ в случае известных f, a можно разрешать относительно неизвестной x, используя обозначение "слой" для прообраза элемента: $\forall_{afx}(x \in \text{слой}(f,a) \leftrightarrow a = f(x))$. Прием блокируется, если в контексте замены имеется равенство, определяющее f через описатель "отображение". В таких случаях будет применяться прием, подставляющий явное выражение для $f(x)$.

Попытка группировки неизвестных

Преобразование предыдущего приема может оказаться полезным, если оба выражения f, a содержат неизвестные. Тогда группировка их в общем подвыражении "слой(f,a)" иногда может упростить задачу. Для проведения упрощений используется нормализатор "уравнпринадлежит", примененный ко всей заменяющей (левой) части эквивалентности. Фильтр "длина(контрольглубины)" контролирует наличие упрощения.

Параметрическое описание семейства функций, принимающих на заданном подмножестве своей области определения заданные значения

Если значения неизвестной функции f на заданном подмножестве C ее области определения выражены через прочие неизвестные, то можно использовать параметрическое описание, выражающее f в виде объединения двух компонент, одна из которых определена на C, а другая - на дополнении к C. Для такого перехода созданы два похожих приема со следующими теоремами:

$\forall_{ABCfg}(C = \mathrm{set}_x(x \in B$ & $A(x)) \to f -$ функция & $\mathrm{Dom}(f) = B$ & $\forall_x(A(x) \to f(x) = g(x)) \leftrightarrow \exists_h(h -$ функция & $\mathrm{Dom}(h) = B \setminus C$ & $f = \text{таблица}\{\lambda_x(g(x), x \in C), h\}))$

$\forall_{ABCDfg}(C = \mathrm{set}_x(x \in B$ & $A(x))$ & $g(x) \in D \to \text{Отображение}(f, B, D)$ & $\forall_x(A(x) \to f(x) = g(x)) \leftrightarrow \exists_h(\text{Отображение}(h, B \setminus C, D)$ & $f = \text{таблица}\{\lambda_x(g(x), x \in C), h\}))$.

Оба приема имеют заголовок "параметризация", т.е. предпринимают попытку решить вспомогательную задачу на описание, в которой условия заменяемой (левой) части эквивалентности заменены параметрическим описанием. Роль параметра здесь играет неизвестная функция h, доопределяющая f на дополнении к C. Фильтр "не(

входит(x6 фикс(1 5 2)))" проверяет, что выражение $g(x)$, определяющее значения f на подмножестве C, не содержит переменной f. Само множество C определяется в антецеденте как пересечение области определения f с множеством значений x, удовлетворяющих антецедентам $A(x)$ кванторного тождества для f. Это пересечение задано описателем "класс", и для получения явного его выражения применен нормализатор "нормкласс". Указатель "внешнийквантор(...)" блокирует попытки идентификации кванторного тождества для $f(x)$ как отрицания квантора существования.

Извлечение информации об элементе набора при решении задачи на доказательство

Если в условии задачи на доказательство встречается выражение $f(a)$, причем некоторая посылка этой задачи представляет собой кванторную импликацию, дающую информацию о значениях функции f, то предпринимается попытка вывода из нее следствия, содержащего подвыражение $f(a)$. Теорема приема имеет вид:

$$\forall_{afPmn}(a \in \{m, \ldots, n\} \ \& \ \forall_i(i \in \{m, \ldots, n\} \to P(f(i))) \to P(f(a)))$$

Указатель "контрольвывода(значение(x6 x1))" определяет инициализацию применения приема при обнаружении выражения $f(a)$, причем фильтры "условие", "тип(доказать)" говорят, что его вхождение должно относиться к условию задачи на доказательство. Первый антецедент обрабатывается проверочным оператором. Переменная P в консеквенте кванторной посылки выделена указателем "отображение", Это означает, что $P(\ldots)$ будет идентифицироваться с произвольным термом. Так как к моменту идентификации консеквента термы f, i уже идентифицированы, то можно использовать указатель "новаргумент(x40 x9 фикс)". Он определит выделение в идентифицирующем для $P(\ldots)$ терме всех вхождений выражения $f(i)$ и проверку того, что i не встречается вне данных вхождений. Таким образом, переход к $P(f(a))$ при выводе следствия будет корректен.

Исключение квантора

Прием усматривает существование функции, принимающей в заданной точке заданное значение. Его теорема имеет вид: $\forall_f(\exists_y(y - \text{функция} \ \& \ y(x) = f(x)))$. Заголовок - "второйтерм", т.е. утверждение, идентифицированное с квантором существования, заменяется на константу "истина". Переменная f выделена указателем "отображение", т.е. $f(x)$ идентифицируется с термом произвольного вида.

Исключение описателя "отображение"

Если в описателе "отображение" для указания значения определяемой им функции в текущей точке x используется выражение "значение(f x)", а условие на x совпадает с условием принадлежности области определения функции f, то данный описатель заменяется на f. Теорема приема такова:

$$\forall_f(\lambda_x(f(x), x \in \mathrm{Dom}(f)) = f).$$

Указатель "смравно(фикс(0 1 2 2))" дает возможность идентифицировать как случай явного подвыражения $\mathrm{Dom}(f)$, так и случай, когда вместо него расположен какой-то терм t, а в контексте имеется равенство $t = \mathrm{Dom}(f)$. Для последовательностей создан еще один аналогичный прием:

$\forall_{fA}$(последовательность$(f, A) \rightarrow \lambda_n(f(n), n -$ натуральное$) = f)$

Упрощение описателя "отображение" при редактировании ответа задачи на описание

Если в ответе задачи на описание появляется равенство, определяющее значение неизвестной x через описатель "отображение", то предпринимается обращение к вспомогательной задаче на упрощение подвыражения, задающего значения описателя. Теорема приема имеет вид:

$\forall_{fAxg}(f(y) = g(y) \rightarrow x = \lambda_y(f(y), A(y)) \leftrightarrow x = \lambda_y(g(y), A(y)))$.

Фильтры "тип(описать)", "цель(редакция)", "корень", "неизвестная(x23)", известно(фикс(0 1 2))" указывают, что выполняется редактирование ответа задачи на описание, причем x - неизвестная. Антецедент выделен указателем "идентификатор", причем его левая часть обрабатывается вспомогательной задачей на преобразование, решаемой до уровня 5 и имеющей дополнительную посылку $A(y)$. Указатель "первыйключ(отображение упростить)" блокирует повторные попытки упрощения того же самого равенства. Указатель "кортежпеременных(x24)" разрешает применение приема для связывающей приставки y произвольной длины.

Определение элемента конкатенации двух наборов

Чтобы определить значение конкатенации двух наборов a, b в i-м разряде, введен прием со следующей теоремой:

$\forall_{abim}(m = l(a) \rightarrow (a; b)(i) = (a(i)$ при $i \leq m,$ иначе $b(i - m)))$.

Напомним, что точка с запятой обозначает в данном случае конкатенацию. Антецедент выделен указателем "идентификатор", и его правая часть обрабатывается нормализатором "нормдлинанабора". В простейших случаях, когда a задано термом "набор", переменной m присваивается числовая константа. Заменяющий терм обрабатывается нормализаторами, позволяющими извлечь в простейших случаях явное выражение для требуемого элемента. В частности, выражения $a(i)$, $b(i-m)$ упрощаются вспомогательными задачами на преобразование, имеющими уровень обращения 2.

Равенство двух описателей "отображение"

Равенство двух описателей "отображение", условия на область определения которых совпадают, преобразуется в кванторное тождество для выражений, указывающих их значения:

$\forall_{fgA}(\lambda_n(f(n), A(n)) = \lambda_m(g(m), A(m)) \leftrightarrow \forall_n(A(n) \rightarrow f(n) = g(n)))$

5.1.1 Равенство значений взаимно-однозначного отображения

Равенство значений взаимно-однозначного отображения преобразуется в равенство значений аргумента:

$\forall_{fab}$(взаимнооднозначно$(f) \rightarrow f(a) = f(b) \leftrightarrow a = b)$

Антецедент выделен указателем "блокпроверок".

Использование кванторного тождества, определяющего значения функции

Если имеется кванторное тождество, определяющее значения функции f на ее области определения, то оно используется для замены $f(t)$ на явное выражение:

$\forall_{fgAt}(\mathrm{Dom}(f) = A \ \& \ \forall_x(x \in A \to f(x) = g(x)) \to f(t) = g(t))$

Фильтр "не(контекст(вид(фикс(2 5 2)значение(x9 x23))))" требует, чтобы заменяющее выражение не имело вида "значение(...)". Это выражение упрощается вспомогательной задачей на преобразование.

Попытка разрешения кванторного тождества относительно значений неизвестной функции

Если кванторное тождество содержит выражение вида $f(x)$, где f - неизвестная функция; x - переменная связывающей приставки тождества, причем данная функция f не встречается в тождестве вне выражений $f(x)$, то предпринимается попытка перейти к тождеству, дающему явное выражение для $f(x)$. Теорема приема имеет вид:

$\forall_{ABCDtf}(f$ – функция & $B(y) = (y = t \ \& \ C \vee \ D) \to \forall_x(A(x) \to B(f(x))) \leftrightarrow \forall_x(A(x) \ \& \ \neg D \to f(x) = t \ \& \ C))$

Заголовок приема - "второйтерм". Заранее неизвестно, где именно в консеквенте $B(f(x))$ заменяемой части теоремы встретится $f(x)$. Поэтому для инициализации приема необходим указатель "контекст(позиция(x2 фикс(0 1 5))вид(x2 значение(x6 x23)))". Он означает, что сначала будет встречено какое-то вхождение в указанном консеквенте выражения $f(x)$. Это идентифицирует переменную f, а также (если она еще не идентифицирована по кванторной приставке) переменную x. Лишь теперь можно будет идентифицировать функциональную переменную $B(\ldots)$. Указатель "новаргумент(x27 x6 фикс)" определяет проверку того, что переменная f будет входить в идентифицирующий для $B(f(x))$ терм T только внутри выражений $f(x)$. После этой проверки функциональная переменная $B(\ldots)$ идентифицируется с шаблоном, полученным из T подстановкой вместо всех вхождений $f(x)$ некоторой переменной X. Чтобы не выбирать новую переменную, в качестве X берется f. Выражение $B(y)$ строится по данному шаблону путем подстановки y вместо X. Оно нужно для явного разрешения тождества относительно $f(x)$. Здесь y - вспомогательная новая переменная. Разрешение $B(y)$ относительно y реализуется вторым антецедентом, который выделен указателем "идентификатор". Левая часть равенства обрабатывается нормализатором - задачей на описание, имеющей неизвестную y и цели "полный", "явное", "прямойответ". Введен ограничитель трудоемкости попытки разрешения - "лимит(200000)". Правая часть антецедента идентифицирует ответ задачи с утверждением $y = t \ \& \ C \ \vee \ D)$, где C, D не содержат y. Последнее обеспечивается фильтрами "не(входит(x24 x29))", "не(входит(x24 x28))". Фильтр "не(заголовок(x29 и))" блокирует применение приема, если заменяющее тождество для значений $f(x)$ может распасться на несколько тождеств из-за дизъюнктивного вида антецедента $\neg D$. Наконец, фильтр "не(контекст(вид(фикс(0 1 5)равно(значение(x5 x23)x4))неизвестная(x5)не(входит(x5 x4))))" проверяет, не является ли рассматриваемое кванторное тождество уже явно разрешенным относительно значений некоторой неизвестной функции x5.

5.2 Приемы символа "функция"

Усмотрение функции

Если в задаче встречается утверждение "функция(t)", причем из контекста удается усмотреть, что значением выражения t служит функция, то данное утверждение заменяется на константу "истина". Для этой цели введены три приема. Первый имеет теорему "родобъекта(функция)" и заголовок "родобъекта". Он предпринимает попытку усмотреть функцию по заголовку выражения t. Второй прием усматривает функцию t из того, что в контексте встречается утверждение "слово(t)". Наконец, третий прием обращается к проверочному оператору "усмфункция". Он имеет теорему $\forall_a(a -$ функция $\rightarrow a -$ функция$)$. Обращение к оператору выполняет антецедент, который выделен указателем "блокпроверок". Указатель "новый" усиливает выполняемое приемом усмотрение из контекста, разрешая периодические повторные его попытки. Первые два приема срабатывают на уровне 0, третий - на уровне 2, причем применяется он только к условиям задачи. Посылки "функция(x)", где x - переменная, лучше сохранять в явном виде, даже если они избыточны. Это упрощает обработку сопровождения по о.д.з.

Ориентация равенства, задающего функцию

Общий прием символа "равно", выполняющий перестановку частей равенства, предусматривает в том числе и ориентацию равенств, имеющих одной из своих частей описатель "отображение". Заметим, что обычно в посылках задач данный описатель помещается в левой части равенства. Это приводит к повсеместной замене его на обозначающую функцию переменную. В рассматриваемом подразделе приведены два приема, корректирующие общие принципы ориентации равенства. Первый из них имеет теорему $\forall_{fgh}(f = \lambda_x(g(x), h(x)) \leftrightarrow f = \lambda_x(g(x), h(x)))$. Он применяется к равенствам, представляющим собой условия задач на описание. f - переменная, а описатель содержит неизвестные. Тогда целесообразно исключение f для получения явных соотношений с неизвестными, и равенство ориентируется по принципу "описатель справа". Второй прием относится к случаям, когда равенство является посылкой задачи на исследование, причем некоторая другая посылка вида "вектпроизводная(...)" содержит переменную f. Здесь реализуется противоположная ориентация.

Усмотрение противоречивых указаний на тип объекта

Прием имеет теорему "родобъекта(функция)" и заголовок "различимы". Если из контекста усматривается, что тип значения выражения t несовместим с типом "функция", то утверждение "функция(t)" заменяется на константу "ложь". Прием использует справочник "родобъекта", усматривающий те одноместные предикаты, которые суть базисные типы объектов, а также справочник "род", указывающий список всех надтипов данного базисного типа.

Условие существования функции, ограничения на которую затрагивают лишь область определения и текущее значение

Утверждение о существовании функции, ограничения на которую сводятся лишь к области определения и к подмножеству, на котором она принимает значения, легко переформулировать без ссылок на функцию. Для такой переформулировки введены следующие два приема:

$\forall_{PQR}(\exists_f(f - \text{функция} \ \& \ P(\mathrm{Dom}(f)) \ \& \ \forall_x(Q(x) \to R(f(x)))) \leftrightarrow \exists_y(P(y)) \ \& \ (\neg(\exists_x(Q(x))) \ \vee \ \exists_x(R(x))))$

$\forall_{PQR}(\exists_f(f - \text{функция} \ \& \ \forall_x(Q(x) \to R(f(x)))) \leftrightarrow \neg(\exists_x(Q(x))) \ \vee \ \exists_x(R(x)))$

В первом из них ограничения имеются как на область определения, так и на множество значений; во втором - только на множество значений. Указатели "новаргумент(x40 x6 фикс)", "новаргумент(x42 x6 фикс)" определяют идентификацию функциональных переменных $P(\mathrm{Dom}(f))$, $R(f(x))$ с такими термами, у которых f встречается, соответственно, только в виде $\mathrm{Dom}(f)$, $f(x)$.

Доказательство равенства функций

Чтобы доказать равенство функций, области определений которых совпадают, предпринимается доказательство равенства значений в произвольной точке этих областей. Предусмотрены два приема:

$\forall_{fg}(f - \text{функция} \ \& \ g - \text{функция} \ \& \ \mathrm{Dom}(f) = \mathrm{Dom}(g) \to (f = g) \leftrightarrow \forall_x(x \in \mathrm{Dom}(f) \to f(x) = g(x)))$

$\forall_{fgAB}(\text{Отображение}(f, A, B) \ \& \ g - \text{функция} \ \& \ \mathrm{Dom}(g) = A \to (f = g) \leftrightarrow \forall_x(x \in \mathrm{Dom}(f) \to f(x) = g(x)))$

Оба приема имеют фильтры "условие", "тип(доказать)", "корень". Уровни срабатывания их равны 4. Антецедент, проверяющий равенство областей определения, выделен указателем "идентификатор"; выражения $\mathrm{Dom}(\ldots)$ обрабатываются нормализатором "нормобласть".

Кванторная расшифровка равенства функций

Если совпадение областей определения функций непосредственно усмотреть не удается, то для доказательства равенства функций предпринимается полная расшифровка этого равенства по определению:

$\forall_{ab}(f - \text{функция} \ \& \ g - \text{функция} \to (f = g) \leftrightarrow \mathrm{Dom}(f) = \mathrm{Dom}(g) \ \& \ \forall_x(x \in \mathrm{Dom}(f) \to f(x) = g(x)))$

Прием применяется, если равенство или его отрицание является условием задачи на доказательство. В первом случае уровень срабатывания равен 6, во втором - 7. Чтобы блокировать обратное преобразование, вводится комментарий "кванторнаясвертка".

Проверочный оператор "усмфункция"

Кроме приема, усматривающего условие "функция(f)" непосредственно в текущем контексте, оператор имеет несколько приемов, усматривающих функции по их частным разновидностям: наборам, последовательностям, случайным величинам, транспозициям, бинарным операциям и т.п.

5.3 Приемы символа "область"

В основном, приемы этого символа собраны в нормализаторе "нормобласть". Ограничимся их рассмотрением.

Таблица

Напомним, что выражение "таблица(A)" позволяет объединять функции множества A, совпадающие друг с другом на пересечениях своих областей определения. Для определения области таблицы служит следующий прием:

$\forall_{ab}(\mathrm{Dom}(\text{таблица}\{a;b\}) = \mathrm{Dom}(a) \cup \mathrm{Dom}(\text{таблица}\{;b\}))$

Заменяющий терм обрабатывается стандартными нормализаторами, так что выражение для области таблицы определяется приемом за один шаг.

Перестановка

$\forall_{fan}(n = \mathrm{card}(a)\ \&\ n - \text{число}\ \&\ \text{перестановка}(f,a) \to \mathrm{Dom}(f) = \{1,\ldots,n\})$

Первый антецедент выделен указателем "идентификатор"; его правая часть обрабатывается нормализатором "нормощность". Второй антецедент, выделенный указателем "блокпроверок", проверяет конечность множества a.

Конечный набор

Определяется область конечного набора - начальный отрезок натурального ряда от 1 до длины набора.

Усмотрение из посылок

Если в контексте имеется посылка вида область(f) $= A$, причем левая часть не входит в A, то преобразуемое нормализатором выражение область(f) заменяется на A.

Использование равенства, определяющего функцию через описатель "отображение"

$\forall_{fgh}(f = \lambda_x(g(x), h(x)) \to \mathrm{Dom}(f) = \mathrm{set}_x(h(x)))$

Антецедент идентифицируется с посылкой нормализатора. Прием применяется лишь в тех случаях, когда нормализатор имеет комментарий "значения". Есть примеры, в которых подстановка явного выражения для области определения функции, извлекаемого из описателя, нецелесообразна.

Последовательность

$\forall_{fA}(\text{последовательность}(f,A) \to \mathrm{Dom}(f) = N)$

Отображение

$\forall_{fAB}(\text{Отображение}(f,A,B) \to \mathrm{Dom}(f) = A)$

Обратная функция

$\forall_{f}(\text{взаимнооднозначно}(f) \to \mathrm{Dom}(\text{обрфункция}(f)) = \mathrm{Val}(f))$

Произведение функций

$\forall_{fg}(\mathrm{Val}(g) \subseteq \mathrm{Dom}(f) \to \mathrm{Dom}(\text{произведение}(f, g)) = \mathrm{Dom}(g))$

Транспозиция

$\forall_{ijk}(\mathrm{Dom}(\text{транспозиция}(i, j, k)) = \{1, \ldots, k\})$

5.4 Приемы символа "образ"

Выражение "образ(f A)" обозначает образ множества A при отображении f.

Образ пустого множества

$\forall_a(\text{образ}(a, \emptyset) = \emptyset)$

Условие пустоты образа

$\forall_{af}(\text{образ}(f, a) = \emptyset \leftrightarrow a = \emptyset)$

Этот и предыдущий приемы выполняют общую стандартизацию; уровень их срабатывания равен 0.

Кванторная свертка в условие включения образа

$\forall_{abf}(\text{образ}(f, a) \subseteq b \leftrightarrow \forall_x(x \in a \to f(x) \in b))$

Применение приема блокируется на этапе редактирования параметрических описаний, используемых в ответе задачи. Указатель "кванторнаясвертка" определяет идентификацию с попыткой выделения свертываемой кванторной импликации из квантора общности по многим переменным, а также из квантора существования. Фильтр "коммент(кванторнаясвертка)" проверяет отсутствие комментария "кванторнаясвертка", означающего, что ранее выполнялось обратное преобразование кванторной развертки. Уровень срабатывания приема равен 0.

Кванторная свертка в условие непересечения с образом

$\forall_{abf}(\text{непересек}(b, \text{образ}(f, a)) \leftrightarrow \forall_x(x \in a \to \neg(f(x) \in b)))$

Кванторная свертка в условие принадлежности образу

$\forall_{abf}(\exists_x(a = f(x) \;\&\; x \in b) \leftrightarrow a \in \text{образ}(f, b))$

Этот прием имеет небольшие отличия от двух предыдущих. Во-первых, появляются два различных уровня срабатывания: при наличии цели "редуцирование" (обычно используемой в работе системы вывода теорем) уровень равен 2, иначе - 0. Кроме того, в отсутствии указанной цели блокируется применение приема к кванторным импликациям.

Кванторная расшифровка условия принадлежности образу

$\forall_{abf}(a \in \text{образ}(f, b) \leftrightarrow \exists_x(a = f(x) \ \& \ x \in b))$

Прием применяется к принадлежности либо отрицанию принадлежности, представляющим собой условие задачи на доказательство. Уровень срабатывания равен 6 (случай отрицания принадлежности) либо 7. В первом случае появляется кванторная импликация, от которой сразу же удается избавиться, перебросив ее антецеденты в посылки задачи. Второй случай менее прост - здесь приходится обращаться к вспомогательной задаче на описание.

Для предотвращения обратного преобразования вводится комментарий "кванторнаясвертка".

Параметрическое описание для условия принадлежности образу

Теорема та же, что у предыдущего приема. Заголовок приема - "параметризация". Принадлежность является условием задачи на описание; левая часть a - неизвестная, не входящая в правую часть. Задача имеет цель "пример" либо "параметризация". Переход к параметрическому описанию предпринимается во вспомогательной задаче, и при успехе ее ответ выдается как результат решения текущей задачи.

Подбор примера

Чтобы подбирать пример множества, образ которого пуст, служит прием, основанный на следующей теореме:

$\forall_{ax}(x = \emptyset \rightarrow \text{образ}(a, x) = \emptyset)$

Уровень срабатывания равен 1. Для применения приема необходимо отсутствие прочих условий, содержащих неизвестную a, кроме, быть может, условия "множество(a)". Чтобы следить за наступлением момента, когда это оказывается выполнено, введен указатель "новый". Тогда попытка применения приема будет происходить даже в случае большого веса текущего условия - при каждом его выходе из "теневой зоны" сканирования.

Образ одноэлементного множества

$\forall_{fx}(\{f(x)\} = \text{образ}(f, \{x\}))$

Преобразование применяется справа налево; уровень срабатывания равен 2.

Образ при тождественном отображении

$\forall_{fA}(\forall_x(x \in A \rightarrow f(x)) \rightarrow \text{образ}(\lambda_x(x, f(x)), A) = A)$

Переменная f является функциональной - она выделена указателем "отображение". Антецедент обрабатывается вспомогательной задачей на доказательство.

Образ при константном отображении

$\forall_{abP}(\text{образ}(\lambda_x(a, P(x)), b) = (\{a\} \text{ при } \exists_y(y \in b \ \& \ P(y)), \text{иначе } \emptyset))$

Заменяющий терм обрабатывается нормализаторами, необходимыми для общей стандартизации. При этом квантор существования упрощается с помощью вспомогательной задачи на преобразование.

Уравнения для образа

1. Разрешение относительно неизвестной условия включения образа.

   $\forall_{abf}(b \subseteq \text{прообраз}(f, a) \leftrightarrow \text{образ}(f, b) \subseteq a)$

   Выражение b содержит неизвестные; выражения a, f - не содержат. Преобразование применяется справа налево. Уровень - срабатывания равен 1 (в особых случаях понижается до 0).

2. Попытка группировки всех неизвестных в одном выражении.

   $\forall_{abc}(b \subseteq \text{Dom}(c) \to \text{образ}(c, b) \subseteq a \leftrightarrow b \subseteq \text{прообраз}(c, a))$

   Выражения a, c содержат неизвестные. Преобразование применяется слева направо. Уровень срабатывания равен 4. Заменяющее утверждение обрабатывается нормализатором "уравнсодержится", предпринимающим попытку разрешить его относительно неизвестных. Успех этой попытки проверяется фильтром "длина(контрольглубины)".

Образ объединения

$\forall_{abcde}(d = \text{образ}(a, b)\ \&\ e = \text{образ}(a, c) \to \text{образ}(a, b \cup c) = d \cup e)$

Антецеденты выделены указателями "идентификатор". Их правые части обрабатываются нормализаторами "нормобраз". Фильтры "не(входит(образ x4))", "не(входит(образ x5))" проверяют, что при упрощении удалось получить явные выражения для образов. Заменяющее выражение получается в результате обработки объединения данных выражений нормализатором "нормобъединение".

Группировка образов при неизвестном отображении

$\forall_{abc}(\text{образ}(c, a) \cup \text{образ}(c, b) = \text{образ}(c, a \cup b))$

Преобразование применяется слева направо. Выражение c содержит неизвестные; выражения a, b - не содержат.

Попытка преобразования к виду объединения пересечений для упрощения выражения

$\forall_{c}(c = \text{образ}(a, b) \to \text{образ}(a, b) = c)$

Антецедент служит для обработки выражения образ(a, b) нормализатором приведения к стандартной форме "стандобъединение". Фильтр "короче(результат теквхожд)" проверяет, что после обработки получается более короткое выражение. Блокируются попытки применения приема к фрагменту выражения, построенного с помощью операций алгебры множеств - обрабатывается лишь само это выражение. Чтобы не предпринимались попытки повторного упрощения результата R, он сохраняется в комментарии (стандобъединение R).

Следствие условий принадлежности множеству и включения его образа

$\forall_{abf}(x \in a\ \&\ \text{образ}(f, a) \subseteq b \to f(x) \in b)$

Антецеденты идентифицируются с посылками задачи на доказательство. Проверяется, что выражение $f(x)$ входит в условие этой задачи, и выводится следствие $f(x) \in b$.

Выделение внешней одноместной операции

Следующий прием находит образ множества A при отображении, полученном последовательным применением отображений g и f:

$\forall_{fghAB}(B = \text{образ}(\lambda_x(g(x), h(x)), A) \rightarrow \text{образ}(\lambda_x(f(g(x)), h(x)), A) = \text{образ}(\lambda_x(f(x), \text{одз}(f(x))), B))$

Отображение f идентифицируется как корневая операция в формуле, задающей составное отображение. Эта операция может быть многоместной, однако лишь единственный ее операнд T должен содержать переменную x. Чтобы получить выражение, определяющее значение $f(x)$, данный операнд заменяется на переменную x. Такой режим идентификации задается указателем "внешфункция(x6 x23)". Антецедент определяет образ B множества A при отображении g. Его правая часть обрабатывается нормализатором "нормобраз". Фильтр "не(входит(образ x27))" проверяет, что удалось найти явное выражение для B. Фильтр "не(заголовок(значение(x7 x23)x23))" блокирует применение приема в вырожденном случае тождественного отображения g: здесь могло бы возникнуть зацикливание.

Определение образа путем решения вспомогательной задачи на описание

Чтобы найти образ множества a при отображении f, можно попытаться явно разрешить уравнение $y = f(x), x \in a$ с неизвестной x и параметром y, и далее извлечь из ответа условие существования для x. Прием, реализующий эти действия, имеет следующую теорему:

$\forall_{fgaP}((x \in a \,\&\, g(x) \,\&\, y = f(x) \,\&\, y-\text{число}) = P(x, y) \,\&\, f(x)-\text{число} \rightarrow \text{образ}(\lambda_x(f(x), g(x)), a) = \text{set}_y(\exists_x(P(x, y))))$

Левая часть первого антецедента представляет собой указанное выше уравнение. Она обрабатывается вспомогательной задачей на описание, имеющей максимальный уровень 6. Цели задачи - "полный", "явное", "прямойответ", "одз", "упростить". "неизвестные x", "параметры x". Ответ задачи идентифицируется с функциональной переменной $P(x, y)$. Явная разрешенность утверждения $P(x, y)$ относительно x обеспечит возможность последующего исключения квантора существования в утверждении $\exists_x(P(x, y))$. Указатель "смравно(фикс(0 1 1))" обеспечивает идентификацию описателя "отображение(...)" в том случае, если он не расположен непосредственно под символом "образ", а вынесен в отдельное равенство. Заметим, что условие y – число добавлено во вспомогательной задаче благодаря второму антецеденту, обрабатываемому проверочным оператором. Таким образом, область действия приема ограничена лишь числовыми функциями. Рекомендуется в качестве самостоятельного упражнения обобщить прием, распространив его на функции с произвольным типом значений.

Образ промежутка

Для основных операций элементарной алгебры приведены приемы, позволяющие непосредственно определять образ промежутка. Например, для умножения:

$\forall_{abcdef}(0 < a \rightarrow \text{образ}(\lambda_x(ax, f(x)), [b, c]) = [ab, ac])$

Хотя в формульной прорисовке указан замкнутый промежуток $[b, c]$, в действительности теорема приема предусматривает произвольные случаи для границы промежутка. В этом легко убедиться, перейдя к текстовой прорисовке. Здесь терм для промежутка имеет вид "промежуток(x2 x3 x4 x5)".

Нормализатор "нормобраз"

Нормализатор включает ряд приведенных выше простых приемов нахождения образа (в частности, в нем продублированы приемы определения образа промежутка).

Усмотрение отображения "на" из взаимной однозначности и равенства конечных мощностей области определения и области значения

$\forall_{ABf}(\mathrm{card}A = \mathrm{card}B$ & конечное(A) & Отображение(f, A, B) & взаимнооднозначно$(f) \to$ образ$(f, A) = B)$

Первый антецедент выделен указателем "идентификатор", прочие - идентифицируются непосредственно.

Усмотрение взаимной однозначности из равенства конечных мощностей области определения и области значения

$\forall_{fABC}(\mathrm{card}A - \mathrm{card}C = 0$ & конечное$(A) \to$ Отображение(f, A, B) & образ$(f, A) = C \leftrightarrow$ Отображение(f, A, C) & взаимнооднозначно$(f))$

Замена выполняется слева направо. Первый антецедент выделен указателем "идентификатор", второй - "блокпроверок".

Образ множества, заданного параметрически

$\forall_{ABft}($образ$(\lambda_x(f(x), A(x)), \mathrm{set}_y(\exists_z(y = t(z)$ & $B(z)))) = \mathrm{set}_x(\exists_z(x = f(t(z))$ & $B(z)$ & $A(t(z)))))$

5.5 Приемы символа "слой"

Чтобы отличать прообраз элемента a при отображении f от прообраза множества, для него введено специальное обозначение "слой$(f\ a)$"

Принадлежность прообразу элемента

Расшифровка условия принадлежности прообразу элемента является преобразованием общей стандартизации и выполняется следующим приемом:

$\forall_{abx}(x \in$ слой$(f, a) \leftrightarrow a = f(x))$

Замена блокируется лишь для условия задачи на описание, у которого f либо x содержит неизвестную.

Кванторная свертка в условие пустоты прообраза элемента

$\forall_{af}($слой$(f, a) = \emptyset \leftrightarrow \forall_x(\neg(a = f(x))))$

Преобразование применяется справа налево; в случае задачи на доказательство оно блокируется.

Кванторная свертка в условие равенства прообразу элемента

$\forall_{abf}(b - \text{set} \rightarrow b = \text{слой}(f, a) \leftrightarrow \forall_x(a = f(x) \leftrightarrow x \in b))$

Кванторная свертка в условие включения в прообраз элемента

$\forall_{abf}(b \subseteq \text{слой}(f, a) \leftrightarrow \forall_x(x \in b \rightarrow a = f(x)))$

Кванторная свертка в условие включения прообраза элемента

$\forall_{abf}(\text{слой}(f, a) \subseteq b \leftrightarrow \forall_x(a = f(x) \rightarrow x \in b))$

Параметрическое описание различия множества и прообраза элемента

$\forall_{abf}(b - \text{set} \rightarrow \neg(b = \text{слой}(f, a)) \leftrightarrow \exists_x(\neg(a = f(x) \leftrightarrow x \in b)))$

Этот и следующие два приема имеют заголовок "параметризация".

Параметрическое описание условия непустоты прообраза элемента

$\forall_{af}(\neg(\text{слой}(f, a) = \emptyset) \leftrightarrow \exists_x(a = f(x)))$

Параметрическое описание условия невключения прообраза элемента

$\forall_{abf}(\neg(\text{слой}(f, a) \subseteq b) \leftrightarrow \exists_x(a = f(x) \ \& \ \neg(x \in b)))$

Случай взаимно-однозначного отображения

В случае взаимно-однозначного отображения условие принадлежности неизвестного элемента прообразу известного элемента при известном отображении переформулируется с помощью обратной функции:

$\forall_{bxf}(\text{взаимнооднозначно}(f) \ \& \ b \in \text{Val}(f) \rightarrow x \in \text{слой}(f, b) \leftrightarrow x = \text{обрфункция}(f)(b))$

5.6 Приемы символа "прообраз"

Посредством "прообраз(f A)" обозначается прообраз множества A при отображении f.

Прообраз пустого множества

$\forall_a(\text{прообраз}(a, \emptyset) = \emptyset)$

Принадлежность прообразу

$\forall_{afx}(x \in \text{прообраз}(f, a) \leftrightarrow x \in \text{Dom}(f) \ \& \ f(x) \in a)$

Преобразование блокируется для условий задачи на описание, у которых f либо x содержат неизвестные.

Включение прообраза в область определения

$\forall_{ab}(\text{прообраз}(b, a) \subseteq \mathrm{Dom}(b))$

Прием заменяет включения указанного вида на константу "истина".

Кванторная свертка в условие включения прообраза

$\forall_{abf}(\text{прообраз}(f, a) \subseteq b \leftrightarrow \forall_x(x \in \mathrm{Dom}(f)\ \&\ f(x) \in a \rightarrow x \in b))$

Уровень срабатывания равен 0. Фильтры и указатели - стандартные для приемов кванторной свертки.

Кванторная свертка в условие равенства прообразу

$\forall_{abf}(b - \mathrm{set} \rightarrow b = \text{прообраз}(f, a) \leftrightarrow \forall_x(x \in \mathrm{Dom}(f)\ \&\ f(x) \in a \leftrightarrow x \in b))$

Кванторная свертка в условие непересечения с прообразом

$\forall_{abf}(\text{непересек}(b, \text{прообраз}(f, a)) \leftrightarrow \forall_x(x \in b\ \&\ x \in \mathrm{Dom}(f) \rightarrow \neg(f(x) \in a)))$

Свертка в принадлежность прообразу при редактировании ответа

$\forall_{afx}(x \in \text{прообраз}(f, a) \leftrightarrow x \in \mathrm{Dom}(f)\ \&\ f(x) \in a)$

Преобразование применяется справа налево. Созданы две версии приема. Обе они применяются к условиям задачи на описание. Одна, с заголовком "первыйтерм", предназначена для свертки конъюнкции, расположенной внутри одного условия. Другая, с заголовком "заменаусловия(первыйтерм)", применяется к двум различным условиям и заменяет их на одно новое условие. В обоих случаях затрагиваемые приемом утверждения не содержат неизвестных, а задача имеет цели, определяющие тенденцию к компактной переформулировке условий.

Подбор примера

$\forall_{ax}(x = \emptyset \rightarrow \text{прообраз}(a, x) = \emptyset)$

Прием подбирает значение неизвестной x задачи на описание, имеющей цель "пример". Предполагается, что эта неизвестная не имеет вхождений в другие условия, кроме, быть может, условия "множество(x)".

Тождественные преобразования выражений с неизвестными, содержащих прообраз

Следующие приемы позволяют выполнять группировку относительно неизвестных:

$\forall_{abc}(\text{прообраз}(c, a) \cup \text{прообраз}(c, b) = \text{прообраз}(c, a \cup b))$

$\forall_{abc}(\text{прообраз}(c, a) \cap \text{прообраз}(c, b) = \text{прообраз}(c, a \cap b))$

$\forall_{abc}(\text{прообраз}(a, c) \setminus \text{прообраз}(a, b) = \text{прообраз}(a, c \setminus b))$

Все преобразования выполняются слева направо; выражение для функции (в первых двух приемах - c, в третьем - a) содержит неизвестные, а прочие выражения - не содержат. Уровень срабатывания равен 1.

Разрешение относительно неизвестной условия включения в прообраз

$\forall_{abf}(b \subseteq \mathrm{Dom}(f) \to b \subseteq \text{прообраз}(f, a) \leftrightarrow \text{образ}(f, b) \subseteq a)$

Преобразование применяется слева направо. Выражения f, b не содержат неизвестных, а выражение a - содержит.

Усмотрение прообраза

$\forall_{fA}(\mathrm{set}_x(x \in \mathrm{Dom}(f)\ \&\ f(x) \in A) = \text{прообраз}(f, A))$

Преобразование применяется слева направо. Уровень срабатывания равен 2. Какие-либо ограничения отсутствуют.

Свертка в прообраз пересечения при редактировании ответа

$\forall_{aeg}(\text{прообраз}(g, a) \cap \text{прообраз}(g, e) = \text{прообраз}(g, a \cap e))$

Преобразование применяется слева направо при редактировании ответа задачи на описание либо упрощении ответа такой задачи, полученного в ее блоке анализа.

Попытка преобразования к виду объединения пересечений для упрощения выражения

$\forall_{abc}(c = \text{прообраз}(a, b) \to \text{прообраз}(a, b) = c)$

Антецедент реализует последовательные обращения к нормализаторам "стандобъединение", "группмножество", "свертка", которые сначала преобразуют прообраз к виду объединения атомарных выражений, а затем выполняют обратную свертку. Фильтр "короче(результат теквхожд)" проверяет, что при этом достигается уменьшение длины выражения. Фильтр "не(контекст(операнд(x4 теквхожд)символ(x4 объединение пересечение разность прямоепроизведение)))" обеспечивает применение приема лишь к максимальным теоретико-множественным подвыражениям.

Непустота прообраза

$\forall_{fAb}(f(b) \in A \to \neg(\text{прообраз}(f, A) = \emptyset))$

Прием заменяет равенство прообраза пустому множеству на константу "ложь".

Усмотрение пустого прообраза

$\forall_{af}(\text{непересек}(a, \mathrm{Val}(f)) \to \text{прообраз}(f, a) = \emptyset)$

Антецедент выделен указателем "блокпроверок". Так как вероятность его истинности невелика, введен указатель "лимит(20000)", строго лимитирующий трудоемкость проверки.

5.7 Приемы символа "прообр"

Для прообраза элемента введен еще один логический символ. Потребность в нем возникает для взаимно-однозначных отображений, когда множество элементов, преобразуемых в данный элемент, сводится к единственному значению. Это единственное значение, отображаемое посредством f в элемент a, обозначается "прообр(f a)".

Условимся считать, что выражение "прообр(f a)" имеет смысл только для взаимнооднозначного f. В прочих случаях о его значении ничего не известно. Для символа "прообр" введено совсем немного приемов, так как он возникал в задачах редко.

Равенство прообразу элемента

$\forall_{afx}(x = \text{прообр}(f, a) \leftrightarrow a = f(x))$

Преобразование применяется слева направо и блокируется в условиях задачи на описание, если одно из выражений f,x содержит неизвестные.

Определение неизвестного элемента по его образу

$\forall_{fxa}(\text{взаимнооднозначно}(f) \to a = f(x) \leftrightarrow x = \text{прообр}(f, a))$

Выражение x содержит неизвестные, а выражения a, f - не содержат. Антецедент обрабатывается проверочным оператором. Если задача имеет цель "пример", то уровень срабатывания увеличен до 4, иначе он равен 0.

Образ прообраза

$\forall_{fa}(\text{взаимнооднозначно}(f)\ \&\ a \in \text{Val}(f) \to f(\text{прообр}(f, a)) = a)$

5.8 Приемы символа "взаимнооднозначно"

Предикат "взаимнооднозначно(f)" используется для обозначения свойства инъективности отображения f (в различных точках функция принимает различные значения).

Кванторная свертка в условие взаимной однозначности

$\forall_f(\text{взаимнооднозначно}(f) \leftrightarrow \forall_{xy}(f(x) = f(y)\ \&\ x \in \text{Dom}(f)\ \&\ y \in \text{Dom}(f) \to x = y))$

Уровень срабатывания равен 0; условия применимости - такие же, как у других рассмотренных ранее приемов кванторной свертки.

Кванторная расшифровка условия взаимной однозначности

Теорема приема берется из предыдущего пункта. Однако, приемов с этой теоремой два: один - для условия задачи на доказательство, другой - для условия задачи на описание. Уровни срабатывания в обоих случаях достаточно высокие - 6 либо 7 для задач на доказательство и 5 для задач на описание.

Усмотрение взаимной однозначности из пустоты области определения

$\forall_a(\text{Dom}(a) = \emptyset \to \text{взаимнооднозначно}(a))$

Прием выполняет вывод следствий в посылках. Кроме того, имеется прием, заменяющий утверждение "взаимнооднозначно(пустоеслово)" на константу "истина".

Тождественное отображение

$\forall_P$(взаимнооднозначно($\lambda_x(x, P(x))$))

Указатель "кортежпеременных(x23)" делает возможным усмотрение взаимной однозначности тождественных функций, заданных как одноместным, так и многоместным описателями "отображение".

Проверочный оператор

Для проверки взаимной однозначности введен проверочный оператор "усмвзаимнооднозначно". Обращение к нему обеспечивается приемом, срабатывающим на уровне 1 при усмотрении вхождения символа "взаимнооднозначно". В проверочном операторе предусмотрены следующие приемы:

1. Пустое отображение - "взаимнооднозначно(пустоеслово)".

2. Проверка взаимной однозначности набора. Шаг индукции по длине набора выполняется следующим образом:

   $\forall_{ab}$(взаимнооднозначно(b) & $\neg(a \in \{;b\}) \to$ взаимнооднозначно(префикс(a, b)))

   Оба антецедента обрабатываются проверочными операторами.

3. Взаимная однозначность перестановки.

   $\forall_{fA}$(перестановка(f, A) $\to$ взаимнооднозначно(f))

4. Поднабор размещения. Если в наборе попарно различных элементов выделяется поднабор, то его элементы тоже попарно различны:

   $\forall_{anAmb}$(размещение(a, n, A) & $\forall_i(i \in \{1, \ldots, m\} \to b(i) \in \{1, \ldots, n\})$ & взаимнооднозначно($\lambda_i(b(i), i \in \{1, \ldots, m\})$) $\to$ взаимнооднозначно($\lambda_i(a(b(i)), i \in \{1, \ldots, m\})$))

   Указатели "развертка(фикс(2)фикс(3 1)фикс(0 1))" и "отображение(x2)" определяют следующий порядок идентификации. Проверяемое утверждение имеет вид "взаимнооднозначно(набор(значение(a $b(1)$), ...,значение(a $b(m)$)))". Таким образом, b идентифицируется с набором термов длины m. Далее второй антецедент, обрабатываемый проверочным оператором, устанавливает, что все $b(i)$ суть элементы множества $\{1, \ldots, n\}$. Наконец, третий антецедент выполняет рекурсивное обращение к проверке взаимной однозначности набора b.

5. Взаимная однозначность линейного отображения.

   $\forall_{ab}(\neg(a = 0) \to$ взаимнооднозначно($\lambda_x(ax + b, A(x))$))

6. Обратная функция.

   $\forall_f$(взаимнооднозначно(f) $\to$ взаимнооднозначно(обрфункция(f)))

7. Произведение отображений.

   $\forall_{fg}$(взаимнооднозначно(f) & взаимнооднозначно(g) $\to$ взаимнооднозначно(произведение(f, g)))

8. Взаимная однозначность транспозиции.

   $\forall_{ijk}$(взаимнооднозначно(транспозиция(i, j, k)))

9. Усмотрение взаимной однозначности числовой функции, заданной таблицей.

   $\forall_{abn}(\forall_i(i \in \{1,\dots,n\} \to b(i) -$число$) \,\&\, \forall_{ij}(i \in \{1,\dots,n\} \,\&\, j \in \{1,\dots,n\} \,\&\, \neg(i = j) \to \neg(b(i) - b(j) = 0)) \to$ взаимнооднозначно(таблица$\{;\lambda_i(a(i) \to b(i), i \in \{1,\dots,n\}\})\}))$

   Прием проверяет, что все значения функции - числа, и далее для проверки различия этих значений применяет к попарным разностям оператор "усмне0". Указатель "развертка(фикс(0 1 1 1))" определяет идентификацию наборов $a(1),\dots,$ $a(n)$, $b(1),\dots,b(n)$; указатель "блокпроверок(1 2)" - применение проверочных операторов для серийных проверок, задаваемых антецедентами.

Существование взаимно-однозначного отображения

Условие существования взаимно-однозначного отображения, переводящего множество A в множество B, заменяется на условие равенства мощностей этих множеств:

$\forall_{AB}(\exists_f(f -$ функция & взаимнооднозначно(f) & $\mathrm{Dom}(f) = A$ & $\mathrm{Val}(f) = B) \leftrightarrow \mathrm{card}A = \mathrm{card}B)$

Указатель "кванторнаясвертка" усиливает идентификацию квантора (учитываются кванторы существования; допускается группировка подкванторных утверждений по рассматриваемой переменной связывающей приставки).

Равенство образов взаимно-однозначных отображений

Если даны два взаимно-однозначных отображения, имеющие одинаковые области определения и одинаковые множества значений, то равенство их значений в точке a эквивалентно равенству образов дополнений точки a до области определения:

$\forall_{fgAa}(\mathrm{Dom}(f) = \mathrm{Dom}(g)$ & $\mathrm{Val}(f) = \mathrm{Val}(g)$ & взаимнооднозначно(f) & взаимнооднозначно(g) & $\mathrm{Dom}(f) \setminus A = \{a\} \to$ образ$(f, A) =$ образ$(g, A) \leftrightarrow f(a) = g(a))$

Преобразование применяется слева направо. Третий и четвертый антецеденты обрабатываются проверочными операторами; прочие антецеденты выделены указателем "идентификатор".

5.9 Приемы символа "Отображение"

Утверждение "Отображение(f A B)" означает, что f есть функция, область определения которой равна A, а множество значений содержится в B. В приводимых далее приемах антецедент "Отображение(...)" обычно берется непосредственно из контекста.

Усмотрение общей информации о функции

Если в контексте утверждения "функция(f)" содержится утверждение "Отображение(f A B)", то первое утверждение заменяется на константу "истина". Аналогично, если утверждение "Отображение(f A B)" встречается в контексте выражения $\mathrm{Dom}(f)$, то последнее заменяется на A.

Усмотрение отличия значения функции от элемента, не входящего в ее область значений

$\forall_{fABac}(a \in A \ \& \ \text{Отображение}(f, A, B) \ \& \ \neg(c \in B) \rightarrow \neg(f(a) = c))$

Первый и третий антецеденты обрабатываются проверочными операторами.

Свертка либо развертка по определению

Для расшифровки по определению утверждения "Отображение(...)" создан прием, имеющий следующую теорему:

$\forall_{fAB}(\text{Отображение}(f, A, B) \leftrightarrow A = \mathrm{Dom}(f) \ \& \ f - \text{функция} \ \& \ \text{образ}(f, \mathrm{Dom}(f)) \subseteq B)$

Прием применяется к условию задачи на описание; либо A, либо B содержит неизвестные, а f - не содержит. Задача не должна иметь цели "мощность", указывающей на наличие внешней задачи определения мощности множества. В последнем случае расшифровка утверждения "Отображение(...)" нежелательна, так как для этого утверждения создано достаточно большое число комбинаторных приемов. Прием свертки по определениям имеет несколько измененную теорему:

$\forall_{fAB}(\text{Отображение}(f, A, B) \leftrightarrow A = \mathrm{Dom}(f) \ \& \ f - \text{функция} \ \& \ \text{образ}(f, A) \subseteq B)$

Замена выполняется справа налево. Предполагается, что преобразуемая конъюнкция входит в условие задачи на преобразование либо на доказательство.

Простейшие условия существования отображения

В разделе собраны несколько приемов, имеющих заголовок "связка", т.е. выполняющих исключение несущественной неизвестной.

1. Существование отображения в непустое множество.

   $\forall_{AB}(a - \text{set} \ \& \ B - \text{set} \ \& \ \neg(B = \emptyset) \rightarrow \exists_f(\text{Отображение}(f, A, B)) \leftrightarrow \text{истина})$

2. Существование взаимно-однозначного отображения одного множества в другое.

   $\forall_{AB}(A - \text{set} \ \& \ B - \text{set} \rightarrow \exists_f(\text{Отображение}(f, A, B) \ \& \ \text{взаимнооднозначно}(f)) \leftrightarrow \text{card}A \leq \text{card}B)$

3. Существование отображения с заданной мощностью прообраза элемента.

   $\forall_{ABbn}(b \in B \rightarrow \exists_f(\text{Отображение}(f, A, B) \ \& \ \text{card}(\text{слой}(f, b)) = n) \leftrightarrow n \leq cardA)$

4. Существование отображения, отличного от заданного.

   $\forall_{ABf}(\neg(A = \emptyset) \rightarrow \exists_g(\text{Отображение}(g, A, B) \ \& \ \neg(g = f)))$

Пустой прообраз элемента из области значений

Пустота прообраза элемента позволяет сузить область значений, указанную в утверждении "Отображение(...)":

$\forall_{fABb}(b \in B \rightarrow \text{Отображение}(f, A, B) \ \& \ \text{слой}(f, b) = \emptyset \leftrightarrow \text{Отображение}(f, A, B \setminus \{b\}))$

Созданы две версии приема: одна преобразует конъюнкцию, другая - два различных условия задачи на описание.

Прообраз отображения с одноэлементной областью значений

Если область значений отображения одноэлементна, то прообраз единственного элемента этой области равен области определения:

$\forall_{fAb}(\text{Отображение}(f, A, \{b\} \to \text{слой}(f, b) = A)$

Усмотрение отображения одного множества в другое

$\forall_{fAB}(f - \text{функция} \ \& \ \mathrm{Dom}(f) = A \ \& \ \mathrm{Val}(f) \subseteq B \leftrightarrow \text{Отображение}(f, A, B))$

Созданы два приема. Один из них применяется к конъюнкции, расположенной под описателем "класс", причем f - переменная связывающей приставки этого описателя. Такая ситуация типична для комбинаторных задач. Второй прием применяется к группе условий задачи на описание. Здесь f - неизвестная; A, B известны.

Сужение области значений

Если явно указан образ области определения, то предпринимается коррекция области значений:

$\forall_{ABC}(C \subseteq B \ \& \ \text{образ}(f, A) = C \to \text{Отображение}(f, A, B) \leftrightarrow \text{Отображение}(f, A, C))$

Фильтр "не(равно(x27 x28))" блокирует применение приема, если термы B, C совпадают. Первый антецедент обрабатывается проверочным оператором.

Усмотрение константного отображения

Если имеется посылка, указывающая, что множество значений отображения одноэлементно, то утверждение "Отображение(f A B)" заменяется на утверждение о том, что функция f является константой:

$\forall_{ABf}(\mathrm{card}(\text{образ}(f, A)) = 1 \to \text{Отображение}(f, A, B) \leftrightarrow \exists_b(b \in B \ \& \ f = \text{конст}(A, b)))$

Усмотрение принадлежности области значений отображения

$\forall_{ABfc}(\text{Отображение}(f, A, B) \ \& \ c \in A \to f(c) \in B)$

Прием применяется к условию задачи на доказательство. Первый антецедент берется из контекста срабатывания; второй - проверяется с помощью вспомогательной задачи на доказательство.

Отображение множества в себя

$\forall_{fAB}(\text{Отображение}(f, A, \mathrm{Dom}(f)) \leftrightarrow \text{Отображение}(f, A, A))$

Условие различия функций

$\forall_{fgAB}(\text{Отображение}(f, A, B) \ \& \ \text{Отображение}(g, A, B) \ \& \ \neg(f = g) \to \exists_a(a \in A \ \& \ \neg(f(a) = g(a))))$

Прием имеет заголовок "вывод". Он применяется в задачах на исследование для расшифровки условия различия двух отображений. Предполагается, что выражения

f, g не имеют неизвестных. Все антецеденты содержатся в списке посылок. После вывода квантора существования будет срабатывать прием, устраняющий этот квантор и вводящий в рассмотрение некоторую точку a, на которой значения отображений различаются.

Выдача ответа задачи на описание

Если единственным условием на неизвестную функцию f является утверждение "Отображение($f\ A\ B$)", то это утверждение берется как ответ задачи.

5.10 Приемы символа "доопределение"

Результат доопределения функции f значением a на множестве M, не пересекающемся с ее областью определения, обозначается "доопределение($f\ M\ a$)".

Область доопределения

$\forall_{fab}(\mathrm{Dom}(\text{доопределение}(f,a,b)) = \mathrm{Dom}(f) \cup a)$

Прием срабатывает на уровне 0 без каких-либо ограничений.

Значение доопределения

$\forall_{abcd}(\text{доопределение}(a,b,c)(d) = (a(d) \text{ при } d \in \mathrm{Dom}(a), \text{ иначе } c))$

Заменяющий терм обрабатывается нормализатором "нормвариант"; его подвыражение $\mathrm{Dom}(a)$ упрощается вспомогательной задачей на преобразование, решаемой до уровня 3. Условие принадлежности упрощается нормализатором "нормпринадлежит".

Прообраз доопределения

Если то значение b, которым доопределяется функция, не принадлежит ее исходному множеству значений, то прообраз в точке b определяется следующим приемом:

$\forall_{fabAB}(\text{Отображение}(f,A,B)\ \&\ \neg(b \in B) \to \text{слой}(\text{доопределение}(f,a,b),b) = a)$

Второй антецедент здесь обрабатывается проверочным оператором. Если точка, в которой берется прообраз, отличается от нового значения, то используется другой прием:

$\forall_{fabc}(\neg(b = c) \to \text{слой}(\text{доопределение}(f,a,b),c) = \text{слой}(f,c))$

Истинность антецедента проверяется с помощью вспомогательной задачи на доказательство.

Взаимная однозначность доопределения

Если множество, на котором доопределяется функция, одноэлементно, то доопределение может оказаться взаимно-однозначным. Условие его взаимной однозначности сводится к взаимной однозначности исходной функции следующими приемами:

$\forall_{fabAB}(\neg(a \in A)\ \&\ \neg(b \in B)\ \&\ \text{Отображение}(f,A,B) \to \text{взаимнооднозначно}(\text{доопределение}(f,\{a\},b)) \leftrightarrow \text{взаимнооднозначно}(f))$

$\forall_{fABab}(\neg(a \in A) \to$ Отображение(f, A, B) & взаимнооднозначно(доопределение$(f, \{a\}, b)) \leftrightarrow$ взаимнооднозначно(f) & Отображение$(f, A, B \setminus \{b\}))$

Ввод неизвестной для значения неизвестного взаимно-однозначного отображения в заданной точке

Если в условиях задачи на описание фигурирует значение $f(a)$ неизвестного взаимно-однозначного отображения f, не связанное внешним квантором или описателем по переменным выражения a, то f представляется как доопределение в точке a некоторого взаимно-однозначного подотображения g. Это позволяет последующим приемам ввести вспомогательную неизвестную для $f(a)$ и анализировать ее независимо от отображения g.

$\forall_{fABa}(a \in A \to$ Отображение(f, A, B) & взаимнооднозначно$(f) \leftrightarrow \exists_{bg}(b \in B$ & Отображение$(g, A\setminus\{a\}, B\setminus\{b\})$ & взаимнооднозначно(g) & $f =$ доопределение$(g, \{a\}, b)))$

Фактически прием вводит параметрическое описание для неизвестного отображения f через вспомогательное отображение g и значение b в точке a. Выражение a идентифицируется с помощью указателя "контекст(условие(x3)позиция(x4 x3)свобоперанд (x4) вид(x4 значение(x6 x1)))", т.е. из некоторого вхождения в условия задачи выражения "значение(f a)", не связанного внешним квантором или описателем.

5.11 Приемы символа "пустоеслово"

Символ "пустоеслово" обозначает набор нулевой длины, т.е. функцию, определенную на пустом множестве. Приводимые далее два приема выполняют общую стандартизацию, относящуюся к этому символу. Они срабатывают на уровне 0.

Отображение с пустой областью определения

$\forall_{fa}($Отображение$(f, \emptyset, a) \leftrightarrow f =$ пустоеслово

Область значений пустого набора

Val(пустоеслово) $= \emptyset$

5.12 Приемы символа "значения"

Выражение "значения(f)" (в формульной прорисовке - Val(f)) обозначает множество значений, принимаемых функцией f на ее области определения.

Усмотрение множества значений

$\forall_f(f -$ функция $\to$ образ$(f, \mathrm{Dom}(f)) = \mathrm{Val}(f))$

Замена выполняется слева направо. Для случая наборов введена еще одна версия данного приема:

$\forall_{fnA}($кортеж$(f, n, A) \to$ образ$(f, \{1, \ldots, n\}) = \mathrm{Val}(f))$

Определение множества значений путем решения задачи на описание

В общем случае для определения множества значений функции нужно решить вспомогательную задачу на описание. Обращение к ней реализовано в следующем приеме: $\forall_{fga}(a = \mathrm{set}_y(\exists_x(g(x)\ \&\ y - \text{число}\ \&\ y = f(x))) \rightarrow \mathrm{Val}(\lambda_x(f(x), g(x))) = a)$

Антецедент выделен указателем "идентификатор". Он присваивает переменной a искомое множество значений, которое определяется с помощью двух последовательно применяемых нормализаторов. Первый нормализатор - вспомогательная задача на описание, разрешающая утверждения под квантором существования относительно переменной x. Второй нормализатор - вспомогательная задача на преобразование, упрощающая результирующее выражение "класс(...)".

Расшифровка принадлежности множеству значений

Для такой расшифровки предусмотрены несколько приемов, в зависимости от типа функции и способа ее задания. Если функция задана через описатель "отображение", то используется следующий прием:
$\forall_{fgy}(y \in \mathrm{Val}(\lambda_x(f(x), g(x))) \leftrightarrow \exists_x(y = f(x)\ \&\ g(x)))$

Утверждения под квантором существования явно разрешаются относительно x, как и в предыдущем приеме. В случае числовых функций прием блокируется. Здесь используется другой прием, аналогичный приведенному выше, но имеющий в консеквенте дополнительный член "y – число". Однако, и он применяется лишь для расшифровки условий принадлежности, являющихся антецедентами кванторных импликаций. Еще два приема, но уже без обработки вспомогательныи задачами, применяются для последовательностей и наборов:

$\forall_{fab}(\text{последовательность}(f, b) \rightarrow a \in \mathrm{Val}(f) \leftrightarrow \exists_n(n - \text{натуральное}\ \&\ a = f(n)))$

$\forall_{xan}(a - \text{слово}\ \&\ n = l(a) \rightarrow x \in \mathrm{Val}(a) \leftrightarrow \exists_i(i \in \{1, \ldots, n\}\ \&\ x = a(i)))$

Расшифровка включения в множество значений

Кванторная импликация, возникающая при расшифровке, предствляется в виде отрицания квантора существования:

$\forall_{afg}(a \subseteq \mathrm{Val}(\lambda_x(f(x), g(x))) \leftrightarrow \neg(\exists_y(y \in a\ \&\ \neg(\exists_x(y = f(x)\ \&\ g(x))))))$

Такое представление позволяет обратиться к вспомогательной задаче на описание, разрешающей подкванторное утверждение квантора $\exists_y$ относительно y. Другая вспомогательная задача разрешает относительно x подкванторное утверждение квантора $\exists_x$. Прием применяется к условию задачи на описание, причем требуется, чтобы выражение Val(...) содержало неизвестные.

Множество значений набора

Конечный набор a представляет собой функцию, определенную на начальном отрезке натурального ряда. Чтобы получить ее множество значений, достаточно применить к a одноместную операцию "перечень":

$\forall_a(\mathrm{Val}(a) = \{; a\})$

Прообраз надмножества множества значений

Такой прообраз равен области определения:

$\forall_{af}(\mathrm{Val}(f) \subseteq a \to \text{прообраз}(f, a) = \mathrm{Dom}(f))$

Антецедент обрабатывается проверочным оператором.

Существование функции с заданными областью определения и множеством значений

$\forall_{AB}(\exists_f(f - \text{функция} \; \& \; \mathrm{Dom}(f) = a \; \& \; \mathrm{Val}(f) = B) \leftrightarrow \mathrm{card}B \leq \mathrm{card}A)$

Нормализатор "нормзначения"

В разделе собраны несколько простых приемов, определяющих множество значений. Множество значений функции, обратной к данной, заменяется на область определения последней; находятся множества значений перестановки, транспозиции и произведения функций f, g (последнее - для случая, когда множество значений g содержит область определения f).

5.13 Приемы символа "конст"

Константная функция, определенная на множестве A и принимающая в любой его точке значение b, обозначается "конст(A b)".

Значение константной функции

$\forall_{abc}(\text{конст}(a, b)(c) = b)$

Область определения константной функции

$\forall_{ab}(\mathrm{Dom}(\text{конст}(a, b)) = a)$

Прообраз элемента

$\forall_{ab}(\text{слой}(\text{конст}(a, b), b) = a)$

$\forall_{abc}(\neg(a = c) \to \text{слой}(\text{конст}(a, b), c) = \emptyset)$

Образ множества

$\forall_{ABa}(A \subseteq B \; \& \; \neg(A = \emptyset) \to \text{образ}(\text{конст}(B, a), A) = \{a\})$

Антецеденты обрабатываются проверочными операторами.

Уравнение для значения константной функции

Если константная функция известна, то для определения ее неизвестного константного значения используется произвольный элемент области определения, выделяемый с помощью символа "элемент":

$\forall_{Afx}(\neg(A = \emptyset) \to f = \text{конст}(A, x) \leftrightarrow x = f(\text{элемент}(A)))$

Преобразуемое равенство является условием задачи на описание. Выражения A,f не содержат неизвестных, а x содержит. Так как прием понадобился лишь в ситуации, когда достаточно было установить факт однозначого определения неизвестных по известным параметрам, дополнительно введен фильтр "цель(функционально)".

5.14 Приемы символа "См"

Выражение "См(a b)" используется для обозначения функции, определенной на единственном элементе a и принимающей на нем значение b. Формульный редактор прорисовывает это выражение как $a \ \rightarrow \ b$, причем для ввода стрелки используется не клавиша "курсор вправо", а клавиша "Ctr-курсор вправо". Выражения данного вида применяются при табличном задании функций с конечной областью определения (см. ниже).

Область определения

$\forall_{ab}(\mathrm{Dom}(a \ \rightarrow \ b) = \{a\})$

Значение в точке

$\forall_{ab}((a \ \rightarrow \ b)(a) = b)$

Прообраз значения

$\forall_{ab}(\text{слой}(a \ \rightarrow \ b, b) = \{a\})$

5.15 Приемы символа "таблица"

Если A - множество функций, согласованных друг с другом на пересечениях областей определения, то выражение "таблица(A)" задает функцию, область определения которой равна объединению областей определений функций множества A, а значения равны значениям этих функций. Обычно выражение "таблица" используется для конечной системы функций, заданных выражениями "См(a_i b_i)". В формульной прорисовке такое выражение имеет вид "таблица$\{a_1 \ \rightarrow \ b_1, \ldots, a_n \ \rightarrow \ b_n\}$". Оно определяет функцию, определенную в точках $a_1, \ldots, a_n$ и принимающую в них значения $b_1, \ldots, b_n$.

Область таблицы

$$\forall_f(\mathrm{Dom}(\text{таблица}(f)) = \bigcup_{x, x \in f} \mathrm{Dom}(x))$$

Замена выполняется слева направо без каких-либо ограничений.

Значение таблицы

$\forall_{abc}(c \in \mathrm{Dom}(a) \rightarrow (\text{таблица}\{a; b\})(c) = a(c))$

Указатель "список(фикс(0 1 1 1 1))" позволяет выбирать тот элемент a, для которого истинен антецедент. Обработка его реализуется проверочным оператором.

Прообраз таблицы

$$\forall_{af}(\text{слой}(\text{таблица}(f), a) = \bigcup_{x, x \in f} \text{слой}(x, a))$$

Заменяющее выражение упрощается вспомогательной задачей на преобразование.

Усмотрение таблицы

Если значение функции задается условным выражением с двумя константными альтернативами, то ее можно преобразовать к виду "таблица(...)":

$\forall_{fgab}(\lambda_x((a \text{ при} f(x), \text{ иначе } b), g(x)) = \text{таблица}\{\text{конст}(\text{set}_x(f(x) \ \& \ g(x)), a), \text{конст}(\text{set}_x(\neg f(x) \ \& \ g(x)), b)\})$

Фактически такое преобразование выполняется лишь в задачах на преобразование, имеющих цель "таблица".

Одноэлементный список

Для отбрасывания избыточной операции "таблица" в случае одноэлементного множества объединяемых функций служит прием:

$\forall_a(a - \text{функция} \rightarrow \text{таблица}\{a\} = a)$

Чтобы не нарушать этим приемом стандартной формы задания многочленов, введен фильтр "не(контекст(подтерм(многочлен(x2 x3 теквхожд))))".

Исключение таблицы при использовании описателя "класс"

$\forall_{AB}(\text{таблица}(\text{set}_f(\exists_i(A(i) \ \& \ f = (i \ \rightarrow \ B(i))))) = \lambda_i(B(i), A(i)))$

Занесение минуса внутрь таблицы

$\forall_{abn}(-\text{таблица}\{; \lambda_i(\text{конст}(a(i), b(i)), i \in \{1, \ldots, n\})\} = \text{таблица}\{; \lambda_i(\text{конст}(a(i), -b(i)), i \in \{1, \ldots, n\})\})$

Указатель "развертка(фикс(0 1 1 1 1)фикс(0 2 1 1))" определяет идентификацию первого описателя "отображение(...)" с термом вида "набор(...)" и создание терма такого же вида для второго описателя.

Сложение таблиц

Если рассматривается сумма двух таблиц, определенных на одном и том же конечном множестве, то выполняется переход к одной таблице:

$\forall_{abcn}(\text{таблица}\{; \lambda_i(\text{конст}(a(i), b(i)), i \in \{1, \ldots, n\})\} + \text{таблица}\{; \lambda_i(\text{конст}(a(i), c(i)), i \in \{1, \ldots, n\})\} = \text{таблица}\{; \lambda_i(\text{конст}(a(i), b(i) + c(i)), i \in \{1, \ldots, n\})\})$

Так как к моменту идентификации второй таблицы точки $a(i)$ ее области определения уже известны, введен указатель "подтерм(фикс(0 1 2 1 1 3 1))", означающий, что эти точки сравниваются с известными значениями.

Равенство функции таблице, содержащей ее подфункцию

Условие равенства функции f таблице, составленной из сужения f на некоторое множество и другой функци g сводится к условию, что g есть сужение f на область определения g:

$\forall_{fgA}(\mathrm{Dom}(g) \subseteq \mathrm{Dom}(f)\ \&\ x \in \mathrm{Dom}(g) \to \text{таблица}\{\lambda_x(f(x), A(x)), g\} = f \leftrightarrow g = \text{сужение}(f, \mathrm{Dom}(g)))$

Указатель "занесениепосылки(2 и(принадлежит(x23 область(x6)) не(значение(x26 x23))))" означает, что при проверке второго антецедента вводятся посылки $x \in \mathrm{Dom}(f), \neg(A(x))$. Таким образом проверяется, что область определения функции f накрывается областями определения функций, упомянутых в таблице.

Вложенные таблицы

Вложенные таблицы устраняются путем объединения классов их подфункций:

$\forall_{ab}(\text{таблица}\{\text{таблица}(a); b\} = \text{таблица}(a \cup \{; b\}))$

Ориентация равенства

В задачах на исследование равенство "$f = \text{таблица}(\ldots)$" ориентируется справа налево: "$\text{таблица}(\ldots) = f$". Это позволяет избежать немедленной подстановки вместо функции f ее табличного задания во всех точках, где эта функция упоминается.

Нормализатор "нормтаблица"

В нормализаторе собраны несколько простых приемов. Если в таблице встречаются две константные функции с одинаковым значением, то они заменяются на констатную функцию, заданную на объединении областей исходных функций. Исключаются вложенные таблицы, а также таблицы с одноэлементным либо пустым списком функций.

Нормализатор "упорядтаблица"

Создан нормализатор "упорядтаблица", переупорядочивающий элементы таблицы $a_1 \to b_1, \ldots, a_n \to b_n$ по возрастанию числовых значений a_i.

5.16 Приемы символа "сужение"

Выражение "сужение(f A)" обозначает сужение функции f на множество A.

Сужение одной функции на область определения другой

Если одна функция, заданная описателем "отображение", сужается на область определения другой функции, также заданной описателем "отображение", то происходит замена условия первого описателя на условие второго (рассмотрение сужения предполагает, что область определения второй функции - подмножество области определения первой).

$\forall_{pqrsf}(f = \lambda_y(r(y), s(y)) \to \text{сужение}(\lambda_x(p(x), q(x)), \mathrm{Dom}(f)) = \lambda_x(p(x), s(x)))$

Нормализатор "нормсужение"

В нормализаторе имеются следующие простые приемы:

1. Сужение функции на конечный список. Реализуется следующее рекурсивное сведение к подсписку:

   $\forall_{abf}(\text{сужение}(f, \{a; b\}) = \text{таблица}\{\text{конст}(\{a\}, f(a)), \text{сужение}(f, \{; b\})\})$

   Чтобы в табличном задании сужения встречались только упрощенные термы для значений функции, выражение $f(a)$ обрабатывается вспомогательной задачей на преобразование.

2. Сужение на пустое множество.

   $\forall_f(\text{сужение}(f, \emptyset) = \text{пустоеслово})$

3. Выделение члена объединения, на котором функция обращается в константу. Если рассматривается сужение функции на объединение множеств, то предпринимается попытка усмотреть такой член объединения, на котором функция есть тождественная константа:

   $\forall_{fghabcd}(f = \lambda_x(g(x), h(x))\ \&\ g(x) = c\ \&\ \text{сужение}(f, b) = d \to \text{сужение}(f, a \cup b) = \text{таблица}\{\text{конст}(a, c), d\})$

   Указатель "контекст(равно(x5 терм(принадлежит(x23 x1))))" присваивает вспомогательной переменной x5 условие принадлежности набора x значений аргументов функции f подмножеству a. Это условие используется как дополнительная посылка при упрощении выражения $g(x)$ во втором антецеденте. Если результат упрощения c не зависит от x, то функция f на a обращается в константу. Третий антецедент реализует шаг рекурсии - определение сужения f на остаток объединения b. Заметим, что условие "принадлежит(x23 x1)" из указателя обрабатывается вспомогательной задачей на описание, разрешающей его явно относительно x.

4. Элиминация прямого произведения. Если рассматривается сужение f на объединение, членом которого служит прямое произведение двух конечных списков, то этот член приводится к виду списка пар.

5.17 Приемы символа "обрфункция"

Если функция f взаимно-однозначна (инъективна), то выражение "обрфункция(f)" обозначает обратную функцию, определенную на множестве значений исходной.

Определение обратной функции путем решения уравнения

$\forall_{fghP}((y = f(x)\ \&\ g(x)) = (x = h(y)\ \&\ P(y)) \to \text{обрфункция}(\lambda_x(f(x), g(x))) = \lambda_y(h(y), P(y)))$

Первый антецедент реализует обращение к вспомогательной задаче на разрешение относительно x условий $y = f(x)$, $g(x)$. Это позволяет найти выражение $h(y)$, определяющее значения обратной функции, а также условие $P(y)$ на принадлежность аргумента y области ее определения. Фильтр "не(заголовок(фикс(0 1 1 3)набор))" и отсутствие указателя "кортежпеременных(x23)" означают, что прием относится к скалярным функциям одной переменной. Случай векторных функций рассматривается в следующем подразделе.

Определение обратной функции путем решения системы уравнений

$\forall_{fghP}((y = f(x)\ \&\ g(x)) = (\forall_i(i \in \{1,\ldots,m\} \to x(i) = h(i))\ \&\ P(y))\ \&\ n = l(f(x))\ \&\ m = l(x) \to \text{обрфункция}(\lambda_x(f(x), g(x))) = \lambda_y(\lambda_i(h(i), i \in \{1,\ldots,m\}), P(y)))$

Указатель "кортежпеременных(x23)" означает, что функция может зависеть от нескольких переменных. Третий антецедент присваивает их количество переменной m. Фильтр "заголовок(фикс(0 1 1 3)набор)" указывает, что выражение $f(x)$, определяющее значения функции f, имеет вид "набор(...)". Второй антецедент присваивает переменной n число элементов данного набора. Как и в третьем антецеденте, здесь используется нормализатор "нормдлинанабора". Фильтр "натуральное(x14)" дает компилятору информацию, что переменная n идентифицирована с натуральной константой. Указатель "переменные(x24 x14)" определяет идентификацию переменной y с набором из n новых переменных. Далее первый антецедент обращается к вспомогательной задаче, разрешающей относительно неизвестных x условия $y = f(x)$, $g(x)$. Таким образом находится система выражений $h(i)$, набор которых дает значения обратной функции. Утверждение $P(y)$ выражает условие принадлежности набора y области определения обратной функции. Указатель "развертка(фикс(1 2 1)фикс(0 2 3))", во-первых, относится к кванторной импликации внутри первого антецедента и означает, что она идентифицируется с конъюнкцией равенств. Во-вторых, он относится к выражению $\lambda_i(h(i), i \in \{1,\ldots,m\})$ и означает, что это выражение строится как "набор($h(1)$, ..., $h(m)$)".

Исключение обратной функции в выражении для мощности множества

Если требуется найти число элементов из области определения обратной функции, удовлетворяющих заданному условию, то предпринимается переформулировка этого числа в терминах исходной функции:

$\forall_{afnA}(x \in \mathrm{Val}(f) \to \mathrm{card}(\mathrm{set}_{xy}(A(x, y, \text{обрфункция}(f)(x)))) = \mathrm{card}(\mathrm{set}_{xy}(A(f(x), y, x)\ \&\ x \in \mathrm{Dom}(f))))$

Первый антецедент проверяет, что условие $A(\ldots)$ влечет принадлежность x области значений функции f. То, что эта функция взаимно-однозначна, вытекает из предположений относительно о.д.з. для операции "обрфункция".

Равенство для значения обратной функции

Равенство, одна из частей которого есть значение обратной функции, переформулируется в терминах исходной функции:

$\forall_{fab}(\text{взаимнооднозначно}(f)\ \&\ a \in \mathrm{Dom}(\text{обрфункция}(f)(a) = b \leftrightarrow b \in \mathrm{Dom}(f)\ \&\ f(b) = a))$

Прием блокируется, если равенство представляет собой условие задачи на описание, выражающее неизвестное выражение b через известные параметры.

Последовательное применение прямого и обратного отображений

$\forall_{fa}(\text{взаимнооднозначно}(f)\ \&\ a \in \mathrm{Dom}(\text{обрфункция}(f)) \to f(\text{обрфункция}(f)(a)) = a)$

Антецеденты обрабатываются проверочными операторами.

Область определения обратной функции

Этот и следующий приемы выполняют общую стандартизацию.

$\forall_f$(взаимнооднозначно$(f) \to$ Dom(обрфункция$(f)) =$ Val(f))

Область значений обратной функции

$\forall_f$(взаимнооднозначно$(f) \to$ Val(обрфункция$(f)) =$ Dom(f))

Функция, заданная таблицей

$\forall_{abn}$(взаимнооднозначно(таблица$\{;\lambda_i(a(i) \ \to \ b(i), i \in \{1,\ldots,n\})\}) \to$ обрфункция(таблица$\{;\lambda_i(a(i) \ \to \ b(i), i \in \{1,\ldots,n\})\}) =$ таблица$\{;\lambda_i(b(i) \ \to \ a(i), i \in \{1,\ldots,n\})\}$)

Указатель "развертка(фикс(1 1 1 1)фикс(0 1 1 1 1)фикс(0 2 1 1))" определяет интерпретацию описателей "отображение" как термов вида "набор(...)".

Нормализатор "нормобрфункция"

В нормализаторе повторяются несколько рассмотренных выше приемов: для определения обратной функции путем решения уравнений и для случая табличного задания функции.

5.18 Приемы символа "перестановка"

Утверждение "перестановка$(f\ A)$" означает, что f есть взаимно-однозначное отображение начального отрезка натурального ряда на конечное множество A.

Представление перестановки объединения двух множеств в виде наложения перестановок этих множеств

Если f - перестановка объединения двух непересекающихся множеств, причем известны позиции, на которых находятся элементы первого множества, то f представляется как результат наложения перестановок каждого множества по отдельности. Заметим, что используемый ниже символ "наложение$(a\ b\ s)$" означает набор, полученный из наборов a,b таким перемешиванием их элементов (с сохранением относительного порядка для каждого поднабора), что элементы набора a оказываются размещены на позициях, номера которых образуют набор s. Нумерация элементов наборов начинается с 1.

$\forall_{fABmn}$(непересек(A,B) & card$A = n$ & взаимнооднозначно$(m) \to$ перестановка$(f,$ $A{\cup}B)$ & $\forall_i(i \in \{1,\ldots,n\} \to f(m(i)) \in A) \leftrightarrow \exists_{gh}(f =$ наложение(g,h,m) & перестановка(g,A) & перестановка(h,B)))

Указатель "развертка(фикс(0 1 2))" определяет идентификацию кванторной импликации с конъюнкцией условий принадлежности множеству A. Указатель "дизъюнктоперанд" блокирует попытки усмотреть заменяемую конъюнкцию в отрицании дизъюнкции.

Усмотрение существования перестановки, имеющей заданные элементы на заданных позициях

Если в задаче на описание имеется несущественная неизвестная f - перестановка на заданном n - элементном множестве, известном либо содержащем лишь отличные от f неизвестные, причем дополнительные условия на перестановку сводятся лишь к фиксации ее значений d в заданных точках c, то предпринимается исключение f. Заменяющие условия накладывают на c, d естественные ограничения: точки c принадлежат промежутку от 1 до n; значения d принадлежат множеству A; количества элементов в наборах c, d равны друг другу и числу равенств для значений f. Последнее означает, что в наборах c, d нет повторений.

$\forall_{Acdnm}(\mathrm{card}A = n \rightarrow (\exists_f(\text{перестановка}(f, A)\ \&\ \forall_i(i \in \{1, \ldots, m\} \rightarrow f(c(i)) = d(i))) \leftrightarrow \{;c\} \subseteq \{1, \ldots, n\}\ \&\ \{;d\} \subseteq A\ \&\ \mathrm{card}\{;c\} = m\ \&\ \mathrm{card}\{;d\} = m))$

Прием имеет заголовок "связка" (исключается несущественная неизвестная f). Указатель "развертка(фикс(0 1 2 2))" определяет идентификацию кванторной импликации с группой равенств - условий задачи.

Декомпозиция перестановки, у которой фиксирована позиция некоторого элемента

$\forall_{fAian}(\mathrm{card}A = n\ \&\ i \in \{1, \ldots, n\}\ \&\ a \in A \rightarrow (\text{перестановка}(f, A)\ \&\ f(i) = a \leftrightarrow \exists_g(\text{перестановка}(g, A \setminus \{a\})\ \&\ f = \text{наложение}(1\ \rightarrow\ a, g, (i)))))$

Хотя в текущей версии приема нет каких-то особых ограничений на контекст применения, подразумевается, что он используется в задачах на определение мощности множеств.

Усмотрение перестановки

Если функция f явно задана с помощью описателя "отображение", то можно попытаться усмотреть истинность утверждения "перестановка(f A)" и заменить его на константу "истина". Для этой цели создан следующий прием:

$\forall_{fAn}(\text{взаимнооднозначно}(f)\ \&\ \mathrm{Dom}(f) = \{1, \ldots, n\}\ \&\ \mathrm{Val}(f) = A \rightarrow \text{перестановка}(f, A))$

Заголовок приема - "второйтерм". Фильтр "заголовок(х6 отображение)" усматривает, что функция f задана явно. После этого первый антецедент (выделенный указателем "блокпроверок") проверяет инъективность отображения f, а второй и третий (выделенные указателем "идентификатор") анализируют область определения и множество значений f. Следующий прием аналогичен: значения функции определяются на начальном отрезке $\{1, \ldots, N\}$ натурального ряда кванторной импликацией, функция взаимно-однозначна и мощность ее множества значений равна N. Утверждение "перестановка(...)" преобразуется в условие равенства области определения функции указанному отрезку:

$\forall_{MNfg}(\forall_i(i \in \{1, \ldots, N\} \rightarrow f(i) = g(i))\ \&\ \forall_{ij}(i \in \{1, \ldots, N\}\ \&\ j \in \{1, \ldots, N\}\ \&\ \neg(i = j) \rightarrow \neg(g(i) = g(j)))\ \&\ \forall_i(i \in \{1, \ldots, N\} \rightarrow g(i) \in M)\ \&\ \mathrm{card}M = N \rightarrow \text{перестановка}(f, M) \leftrightarrow \mathrm{Dom}(f) = \{1, \ldots, N\})$

Первый антецедент - кванторная импликация, задающая значения функции f. Здесь $g(i)$ - функциональная переменная, идентифицируемая с термом, определяющим значение $f(i)$. Второй и третий антецеденты обрабатываются вспомогательными задачами на доказательство, четвертый выделен указателем "идентификатор". Указатель "лимит(...)" делает попытку применения приема кратковременной.

Перестановка на одноэлементном множестве

Если известно, что областью действия перестановки является одноэлементное множество, то она представляется как одноэлементный набор, содержащий единственный элемент этого множества:

$\forall_{af}(\mathrm{card}a = 1 \to \text{перестановка}(f, a) \leftrightarrow f = (\text{элемент}(a)))$

Сужение перестановки

Если имеется инъективное отображение, то условие, что его сужение на начальный отрезок натурального ряда образует перестановку заданного множества, заменяется на условие равенства данного множества образу начального отрезка. Введены две версии приема, реализующего это преобразование:

$\forall_{fnA}(n$ – натуральное & $\{1, \ldots, n\} \subseteq \mathrm{Dom}(f)$ & взаимнооднозначно$(f) \to$ перестановка(сужение$(f, \{1, \ldots, n\}), A) \leftrightarrow$ образ$(f, \{1, \ldots, n\}) = A)$

$\forall_{fAmx}$(перестановка(f, A) & $0 \leq \mathrm{card}A - m \to$ перестановка(сужение$(f, \{1, \ldots, m\})$, $x) \leftrightarrow x =$ образ$(f, \{1, \ldots, m\}))$

5.19 Приемы символа "тождфунк"

Посредством "тождфунк(A)" обозначается тождественная функция, определенная на множестве A. Приводимые ниже простые приемы выполняют общую стандартизацию и срабатывают на уровне 0.

Значение тождественной функции

$\forall_{ab}(\text{тождфунк}(a)(b) = b)$

Область определения тождественной функции

$\forall_{a}(\mathrm{Dom}(\text{тождфунк}(a)) = a)$

Прообраз элемента

$\forall_{ab}(\text{слой}(\text{тождфунк}(a), b) = \{b\})$

5.20 Приемы символов "выборка", "Вставка", "кортеж", "кортежпар"

Перечисленные символы относятся к конечным наборам. Для них было создано лишь несколько простых приемов, так что мы ограничимся здесь краткой справкой. Посредством "выборка$(a\ B)$" обозначается поднабор набора a,получающийся отбрасыванием всех разрядов, на которых расположены элементы, не принадлежащие множеству B. Выражение "Вставка$(a\ i\ c)$" обозначает результат вставки элемента c в набор a, после которой этот элемент занимает i - й разряд набора. Утверждение "кортеж(a n A)" означает, что a есть набор длины n, составленный из элементов множества A. Наконец, выражение "кортежпар(a b)" обозначает набор пар, первые элементы которых образуют набор a, а вторые - набор b. С приемами, созданными для этих символов, рекомендуется ознакомиться самостоятельно.

5.21 Приемы символа "последовательность"

Утверждение "последовательность(f A)" означает, что f есть функция, определенная на множестве натуральных чисел и принимающая значения на множестве A.

Область определения последовательности

$\forall_{fa}(\text{последовательность}(f,a) \to \mathrm{Dom}(f) = \mathbf{N})$

Усмотрение ответа

Если в задаче на описание требуется найти последовательность x, то одной из возможных форм ответа является указание кванторного тождества, задающего общий вид члена последовательности и сопровождаемого утверждением "последовательность(x a)". Прием, усматривающий найденный ответ такого вида, имеет следующую теорему:

$\text{последовательность}(x,a)\ \&\ \forall_n(n - \text{натуральное} \to x(n) = f(n))$

Заголовок приема - "ответзадачи". Указатель "смответ(2)" фиксирует выбор точки привязки приема в кванторном тождестве. Указатель "внешнийквантор(фикс(0 2))" блокирует попытки идентифицировать кванторную импликацию с отрицанием квантора существования.

Постояннная последовательность

Условие "последовательность(f A)" для константной последовательности f заменяется на условие принадлежности ее общего члена множеству A:

$\forall_{aA}(\text{последовательность}(\lambda_n(a, n - \text{натуральное}, A) \leftrightarrow a \in A))$

Усмотрение последовательности с помощью проверочного оператора

Для проверки того, что f является последовательностью элементов заданного множества, создан проверочный оператор "усмпоследовательность" (см. ниже). Обращения к этому оператору для усмотрения истинности утверждения "последовательность(...)" выполняет следующий прием, антецедент которого выделен указателем "блокпроверок":

$\forall_{ab}(\text{последовательность}(a,b) \to \text{последовательность}(a,b))$

Усмотрение первого элемента, обладающего заданным свойством

Если нужно установить, что неизвестный номер n элемента известной последовательности определяется по данной последовательности однозначно, может оказаться полезным рассмотрение некоторого свойства, которым n обладает, а меньшие номера - не обладают. Тогда n однозначно находится как наименьший номер элемента, обладающего этим свойством. Теорема приема имеет вид:

$\forall_{ABCpn}($последовательность(p, B) $\&$ $p(n) \in A$ $\&$ $\forall_j(j \in \{1, \ldots, n-1\}\neg(p(j) \in A)) \rightarrow$ $n = \inf(\mathrm{set}_m(p(m) \in A$ $\&$ $m -$ натуральное$)))$

Заголовок приема - "выводусловия". Роль свойства играет принадлежность n множеству A,которая должна быть явно представлена в контексте. Фильтр "цель(функционально)" указывает, что в задаче достаточно установить факт однозначного восстановления неизвестного номера n по последовательности p. Третий антецедент проверяется при помощи вспомогательной задачи на доказательство.

Получение явного выражения для общего члена последовательности путем разрешения кванторного тождества, содержащего этот общий член

$\forall_{ABMfg}($последовательность(f, M) $\&$ $(A(x)$ $\&$ $x \in M) = (x = g(n)$ $\&$ $B(n)) \rightarrow \forall_n(n -$ натуральное $\rightarrow A(f(n))) \leftrightarrow f = \lambda_n(g(n), n -$ натуральное$)$ $\&$ $\forall_n(n -$ натуральное $\rightarrow$ $B(n)))$

Преобразуемое кванторное тождество находится в левой части эквивалентности. Первый антецедент берется из контекста замены; он позволяет идентифицировать последовательность f. Указатель "новаргумент(x26 x6 фикс)" дает возможность идентифицировать шаблон $A(f(n))$ путем рассмотрения всех вхождений выражения $f(n)$ в кванторное тождество. Далее указатель "новаяпеременная(x23)" вводит вспомогательную переменную x, и начинается обработка второго антецедента. Здесь происходит разрешение относительно неизвестной x условий $A(x), x \in M$. Если результат содержит явное выражение $g(n)$ для x, то формируется заменяющий терм. Первый член его - явное задание последовательности с помощью описателя "отображение"; второй член - кванторная импликация, содержащая остаточную информацию, предоставляемую преобразуемым кванторным тождеством.

Проверочный оператор "усмпоследовательность"

Проверочный оператор использует несколько простых приемов - усмотрение последовательности в подпоследовательности; усмотрение последовательности в функции натурального аргумента, заданной описателем "отображение", и усмотрение последовательности, полученной с помощью операции "кортежи" как результат выписывания подряд заданной последовательности конечных фрагментов.

5.22 Прочие символы

В разделе "функции" рассматривается еще ряд понятий, вводившихся для поддержки задач других разделов. Число связанных с этими понятиями приемов весьма мало. Однако, они фиксируют относящийся к функциям логический язык решателя, и поэтому заслуживают краткого упоминания, хотя бы на уровне определений.

Утверждение "слово(a)" означает, что a есть функция, определенная на начальном отрезке натурального ряда (в том числе пустом). Фактически, таким образом в решателе обозначаются произвольные конечные наборы. Утверждение "подпоследовательность$(p \quad q)$" означает, что p есть подпоследовательность последовательности

q. Выражение "кортежи(p)" определяет последовательность, полученную выписыванием подряд конечных наборов, образующих последовательность p. Утверждение "размещение(a n A)" аналогично утверждению "кортеж(a n A)", однако на составленный из элементов множества A набор a длины n накладывается то дополнительное ограничение, что разряды попарно различны. Выражение "возрнабор(a)" обозначает результат переупорядочения по неубыванию числового набора a. Аналогично определяется выражение "убывнабор(a)". Выражение "список(f)" обозначает набор, перечисляющий значения функции f, определенной на конечном множестве чисел, в порядке возрастания аргументов. Выражение "индикатор(A B c)" обозначает функцию, определенную на множестве A и принимающую на пересечении его с B значение c, а вне B - значение 0. Выражение "сумма(a R)" позволяет суммировать конечный набор a функций, принимающих значения в кольце R. Областью определения суммы считается объединение областей определения слагаемых, причем значение в точке определяется только по тем слагаемым, которые в ней определены. Выражение "констнабор(a n)" обозначает конечный набор длины n, на каждой позиции которого расположен один и тот же элемент a.

Выражение "суперпоз(f i g)" обозначает результат подстановки функции g вместо i - го аргумента функции f. Набор аргументов новой функции получается конкатенацией набора аргументов функции f с пропущенной i - й переменной, и набора аргументов функции g. Выражение "переименование(f p)" обозначает функцию, получаемую из многоместной функции f при переименовании (возможно, с отождествлением) ее аргументов, определяемым набором p. Длина набора p равна числу аргументов функции f, причем на i - м месте расположен тот новый номер, который i - й аргумент получает после переименования. Выражение "произведение(a)" обозначает произведение функций, образующих конечный набор a. Область определения произведения состоит из элементов области определения последней функции набора a, для которых определены все промежуточные значения. Утверждение "переводит(p a b)" означает, что p есть конечный набор функций, последовательное применение которых к элементу a дает элемент b. Выражение "функстепень(f n)" обозначает функцию, значение которой получается n - кратным последовательным применением функции f.

5.23 Упражнения

Логические символы рассмотренного раздела пока слабо обеспечены приемами, и это открывает хорошие возможности для самостоятельного развития решателя. Мы приведем ниже лишь несколько простых примеров, иллюстрирующих этот процесс. Читателю предоставляется самостоятельно продолжить работу в данном направлении.

1. Ввести приемы, позволяющие решить задачу на описание, в которой требуется привести пример взаимно-однозначного отображения f, определенного на множестве $\{1, 2\}$ и принимающего значения на множестве $\{3, 4\}$.

2. Ввести приемы, позволяющие найти функцию f, сужение которой на множество $\{1, 2\}$ определяется таблицей $\{1 \ \rightarrow \ 3, 2 \ \rightarrow \ 4\}$, сужение на множество $\{3, 4\}$ - таблицей $\{3 \ \rightarrow \ 1, 4 \ \rightarrow \ 2\}$, причем область определения равна $\{1, 2, 3, 4\}$

3. Ввести приемы для решения задачи на описание, имеющей условия "перестановка(f, $\{1, 2, 3\}$)", $f(1) = 2$, $f(3) = 1$ и неизвестную f.

4. Ввести приемы для решения задачи на описание, имеющей условие "выборка((1, 3, 2, 4, 1, 5), x) = (1, 1, 5)" и неизвестную x.
5. Ввести приемы, позволяющие найти набор "наложение((1, 2, 3), (4, 5), (1, 3, 4))".
6. Ввести приемы, позволяющие решить уравнение "Вставка((1, 2, 3, 4),x,y) = (1, 5, 2, 3, 4)".
7. Ввести прием, позволяющий найти набор "кортежпар((1, 2, 3), (4, 5, 6))".
8. Ввести приемы, позволяющие найти последовательность a, удовлетворяющую условиям: "последовательность(a $\mathbf{N}$)", "подпоследовательность($\lambda_i(i, i$ – натуральное), a)", "подпоследовательность($\lambda_i(i^2, i$ – натуральное), a)".
9. Ввести приемы, позволяющие перейти к табличному заданию функции "индикатор($\{1,2,3\},\{1,3\},1$)".
10. Ввести приемы, позволяющие найти явное задание для функции "функстепень($\lambda_i(i^2, i$ – натуральное$), 3$)".

5.23.1 Указания

1. Прежде всего, вводим условия задачи на описание: "взаимнооднозначно(f)", "Dom(f) = $\{1,2\}$", "Val(f) = $\{3,4\}$". Выбираем в оглавлении целевых установок пункт "Найти пример значений неизвестных" и вводим неизвестную f. Убеждаемся, что система не решает данную задачу.

   Если это не так, т.е. при обучении системы, уже после написания данного раздела, пришлось ввести приемы, обеспечивающие решение, то рекомендуется найти такие приемы путем пошаговой трассировки и временно отключить. Это замечание, не повторяя его каждый раз, относим ко всем приводимым в книге упражнениям. Впрочем, по мере возможности, предоставляемое упражнениями поле для самостоятельного творчества читателя будет сохраняться.

   В нашей задаче требуется найти пример взаимно-однозначного отображения одного конечного множества на другое. При этом оба множества заданы путем явного перечисления своих элементов. Последний случай настолько типичен, что для него стоит создать отдельный прием. Оставляя для читателя возможность самостоятельно развить технику установления взаимно-однозначного соответствия между двумя конечными множествами, заданными другими средствами, ограничимся этим приемом. Так как речь идет о подборе примера, теорема приема должна представлять собой кванторную импликацию, консеквентом которой будет конъюнкция реализуемых условий. В нашем случае это условия взаимной однозначности f и равенства областей определения и значений f двум конкретным перечням $\{;a\}$, $\{;b\}$. Антецедентами теоремы приема естественно сделать утверждения $l(a) = n$, $l(b) = n$, указывающие на совпадение длин наборов a, b и вводящие вспомогательное обозначение n, а также утверждения о попарном различии элементов списков a, b. Наконец, в антецеденты заносим равенство, определяющее f с помощью таблицы, где элементу $a(i)$ сопоставлен элемент $b(i)$. Окончательно, теорема приема принимает следующий вид:

$\forall_{abfn}(l(a) = n \;\&\; l(b) = n \;\&\; \text{взаимнооднозначно}(a) \;\&\; \text{взаимнооднозначно}(b)$
$\&\; f = \text{таблица}\{;\lambda_i(a(i) \;\rightarrow\; b(i), i \in \{1, \ldots, n\})\} \rightarrow \text{взаимнооднозначно}(f)$
$\&\; \mathrm{Dom}(f) = \{;a\} \;\&\; \mathrm{Val}(f) = \{;b\})$

Прием имеет заголовок "подборзначений". Указатель "подборзначений(5)" выделяет тот антецедент, который будет заменять конъюнкцию утверждений консеквента. Указатели "идентификатор(1 2)", "блокпроверок(3 4)" задают способ обработки прочих антецедентов. Чтобы переменной n было присвоено численное значение общей длины наборов a, b, выражения $l(a)$, $l(b)$ обрабатываются нормализатором "нормдлинанабора". Указатель "развертка(фикс(5 2 1 1))" необходим, чтобы получить явное представление перечня таблицы через выражение "набор(...)".

Выбираем для приема какой-либо средний уровень срабатывания, например, 4. Прочие фильтры его - "тип(описать)", "условие", "неизвестная(f)", "или(цель(пример)не(цель(полный)))", "заголовок(x1 набор)", "заголовок(x2 набор)". Последние два гарантируют, что значением n будет натуральное число.

После создания приема убеждаемся, что задача решается.

2. Вводим задачу на описание, имеющую условия "сужение(f,$\{1, 2\}$) $= \{1 \;\rightarrow\; 3, 2 \;\rightarrow\; 4\}$", "сужение($f$,$\{3, 4\}$) $= \{3 \;\rightarrow\; 1, 4 \;\rightarrow\; 2\}$", "$\mathrm{Dom}(f) = \{1, 2, 3, 4\}$". В отличие от предыдущего примера, нам нужно найти все значения неизвестной f, для которых истинны условия. Убеждаемся, что система задачу не решает.

   Естественным приемом для решения уравнений вида "сужение(f, A)$=$ g", где A, g известны, а f - неизвестная, является представление f как доопределения известной функции g на дополнении A до области определения f. Это доопределение можно представить в виде "таблица($\{g, h\}$)"; h - функция, имеющая своей областью определения указанное дополнение. Таким образом, приходим к следующей теореме приема:

   $\forall_{fgA}(A \subseteq \mathrm{Dom}(f) \;\&\; \mathrm{Dom}(g) = A \rightarrow \text{сужение}(f, A) = g \leftrightarrow \exists_h(h - \text{функция} \;\&\; f = \text{таблица}\{g, h\} \;\&\; \mathrm{Dom}(h) = \mathrm{Dom}(f) \setminus A))$

   Так как замена эквивалентная, прием имеет заголовок "второйтерм". Чтобы решатель исключил квантор существования, перейдя к вспомогательной неизвестной h, вводим указатель "примечание(серия)". Для уточнения контекста срабатывания вводим фильтры: "уровень(5)"; "тип(описать)"; "условие"; "корень"; "неизвестная(x6)"; "известно(x26)"; "известно(x7)". Чтобы отсечь вырожденный случай, когда области определения функций f, g совпадают, добавляем фильтр "не(равно(x26 терм(область(x6))))". Первый антецедент выделяем указателем "блокпроверок", второй - указателем "идентификатор". Наконец, создаем нормализаторы. В первую очередь, здесь следует обработать нормализатором "нормобласть" выражения для областей определения f, g.

   После создания приема проводим трассировку решения задачи и убеждаемся, что ответ получается. В действительности при рассмотрении данного примера обнаружилось отсутствие ряда других мелких, но необходимых приемов. На данном этапе обучения это неудивительно. Указанные приемы были введены в решатель и оставлены в нем, так как могут быть полезны во многих других случаях. Ситуация, когда основной прием, введенный для решения задачи, оказывается недостаточен, и нужно вводить серию простых сопровождающих приемов, достаточно типична. Читатель должен быть к этому готов, и для большей наглядности мы приведем список тех приемов, которые в нашем примере понадобилось создавать дополнительно.

Прежде всего, нужен прием, усматривающий вырожденное сужение:

$\forall_{fA}(A = \mathrm{Dom}(f) \to \text{сужение}(f, A) = f)$

Прием относится к оператору "нормсужение", хотя его можно было бы продублировать, оформив также как прием сканирования задачи.

Далее понадобился прием, определяющий область определения функции, заданной "одноточечной" таблицей:

$\forall_{ab}(\mathrm{Dom}(a\ \to\ b) = \{a\})$

Этот прием отнесен к нормализатору "нормобласть".

Понадобился прием, отбрасывающий при сужении таблицы на некоторое множество те ее элементы, которые выходят за рамки данного множества:

$\forall_{abcA}(\neg(a \in A) \to \text{сужение}(\text{таблица}\{a\ \to\ b; c\}, A) = \text{сужение}(\text{таблица}\{; c\}, A))$

Этот прием нужен в двух вариантах - как прием сканирования задачи, инициирующий обработку сужения, и как прием нормализатора "нормсужение", используемый при рекурсивных обращениях (заменяющий терм сам обрабатывается нормализатором "нормсужение").

Наконец, понадобилось скорректировать веса ранее созданных приемов пакета "нормсужение", чтобы новые приемы этого пакета срабатывали первоочередным образом.

3. Вводим условия задачи на описание: "перестановка(f $\{1, 2, 3\}$)", "$f(1) = 2$", "$f(3) = 1$". Чтобы решатель мог начать самостоятельные действия по ее решению, нужно перейти к параметрическому представлению f в виде "таблица($\{1\ \to\ b(1), 2\ \to\ b(2), 3\ \to\ b(3)\}$)". Здесь $b(1), b(2), b(3)$ - три новые неизвестные. Теорема приема, обеспечивающего данный переход, имеет вид:

   $\forall_{afn}(\text{перестановка}(f, \{; \lambda_i(a(i), i \in \{1, \dots, n\})\}) \;\&\; \text{взаимнооднозначно}(a) \to \exists_b(f = \text{таблица}\{; \lambda_i(a(i)\ \to\ b(i), i \in \{1, \dots, n\})\}))$

   Прием имеет заголовок "выводусловия". Первый антецедент идентифицируется с условием "перестановка($f, \{1, 2, 3\}$)"; второй - обрабатывается проверочным оператором. Указатель "переменные(x2 x14)" определяет выбор для b набора новых переменных, имеющего ту же длину, что и набор a. Указатель "примечание(серия)" заставит решатель рассматривать новое условие как параметрическое описание для неизвестной f. В результате квантор существования по b будет устранен, и переменные b станут новыми неизвестными. Прочие указатели приема - обычные: "блокпроверок(2)", "отображение(x1)", "развертка(фикс(1 2 1) фикс(0 2 2 1 1))". Фильтры тоже не содержат каких-либо особенностей: "уровень(4)", "тип(описать)", "условие", "неизвестная(x6)", "известно(фикс(1 2))".

   После компиляции приема запускаем пошаговую трассировку. Решатель использует равенства для $f(1), f(3)$, чтобы найти значения неизвестных $b(1), b(3)$. Затем он сводит утверждение вида "перестановка(таблица($\ldots$)A)" к равенствам для области определения и области значений таблицы. Из этих равенств извлекается значение неизвестной $b(3)$.

4. Вводим условие задачи на описание: "выборка$((1,3,2,4,1,5),x) = (1,1,5)$". Очевидно, значением неизвестной x должно служить множество элементов набора в правой части уравнения. При этом необходима проверка:

   $\forall_{abx}(\{;b\} \subseteq \{;a\} \rightarrow \text{выборка}(a,x) = b \leftrightarrow x = \{;b\} \,\&\, \text{выборка}(a,\{;b\}) = b)$

   В этой теореме антецедент не является логически необходимым; он играет роль дополнительного фильтра. Впрочем, его наличие означает, что решателю будет нужен (но не для рассматриваемой задачи) еще один прием - специально для случая, когда включение нарушается. Второй конъюнктивный член консеквента обеспечивает проверку. Прием имеет заголовок "второйтерм" и фильтры "уровень(2)", "тип(описать)", "условие", "известно(x1)", "известно(x2)", "не(известно(x23))". Указатель "блокпроверок(1)" определяет обработку антецедента проверочным оператором.

5. Вводим условие "наложение$((1,2,3),(4,5),(1,3,4))$" задачи на преобразование. Выбираем стандартную целевую установку "упростить в о.д.з.". Чтобы находить наложение двух наборов по заданному списку номеров позиций, которые в результате займет первый набор, создадим новый пакетный нормализатор "нормналожение". Заметим, что хотя третий аргумент операции "наложение$(\ldots)$" - набор, порядок его элементов несуществен.

   Входим в подраздел символа "наложение" оглавления базы приемов и используем интерфейс ввода нового пакетного нормализатора. Нажимаем "Ctr-п" и выбираем левой кнопкой мыши прямоугольник "Нормализатор". Затем нажимаем "с" и вводим название символа "наложение". По завершении ввода нажимаем Enter дважды: первое нажатие завершает редактирование, второе - инициализирует пакет. Далее будем заполнять пакет приемами. Вычисление результирующего набора определим по рекурсии. Начнем с завершающего шага, когда один из наборов пуст. Создаем в оглавлении после автоматически созданного пункта "Переключатель уровня" новый пункт "Пустой набор". В нем размещаем следующие два приема:

   $\forall_{ab}(\text{наложение}(a, \text{пустоеслово}, b) = a)$

   $\forall_{a}(\text{наложение}(\text{пустоеслово}, a, \text{пустоеслово}) = a)$

   Как обычно, при инициализации пакетного нормализатора соблюдаем аккуратность после компиляции первого приема: его немедленная перекомпиляция приведет к порче пакета. Для восстановления пакета понадобится удалить ЛОС-программу данного приема, а также приема "Переключатель уровня" (клавиша F7), после чего снова их откомпилировать, начиная с переключателя уровня. Если в нормализаторе откомпилированы хоты бы два приема, допустима перекомпиляция любого из них.

   Переходим к рассмотрению шага рекурсии. При вычислении значения "наложение$(a\ b\ c)$" рассматриваем два случая: либо набор c содержит единицу, и тогда в начало результирующего набора помещается первый элемент набора a, либо он не содержит единицы, и тогда берется первый элемент набора b. В обоих случаях перед рекурсивным обращением c корректируется: единица, если она есть, отбрасывается, а все прочие элементы уменьшаются на 1. Таким образом приходим к следующим двум теоремам приемов:

   $\forall_{abcden}(1 \in \{;d\} \,\&\, \{;e\} = \{;d\} \backslash \{1\} \,\&\, l(e) = n \,\&\, p = \text{наложение}(b, c, \lambda_i(e(i) - 1, i \in \{1, \ldots, n\})) \rightarrow \text{наложение}(\text{префикс}(a,b), c, d) = \text{префикс}(a,p))$

$\forall_{abcden}(\neg(1 \in \{;d\})\ \&\ l(d)=n\ \&\ p=\text{наложение}(a,c,\lambda_i(d(i)-1, i \in \{1,\dots,n\})) \to \text{наложение}(a,\text{префикс}(b,c),d)=\text{префикс}(b,p))$

Антецеденты $p = \dots$ выделены указателем "идентификатор"; они реализуют рекурсивное обращение (правый части обрабатываются нормализатором "нормналожение"). Первый прием имеет фильтры "заголовок(x4 набор)", "заголовок(x5 набор)", второй - фильтр "заголовок(x4 набор)". Они введены, чтобы компилятор мог правильно сформировать выделенные указателем "развертка(...)" описатели $\lambda_i(\dots)$. Заметим, что в первом приеме введен вспомогательный набор e, полученный отбрасыванием единицы из третьего операнда "наложения", и лишь затем выполняется вычитание единицы из его элементов. Существенной деталью является то, что для проверки вхождения единицы в набор d предпринимается рассмотрение созданного на базе этого набора перечня $\{;d\}$ (множества элементов набора).

Однако, первый прием имеет небольшой изъян - если набор d был одноэлементным, то после отбрасывания из его элементов единицы возникнет пустое множество, и его не удастся идентифицировать как $\{;e\}$: последний терм имеет своим заголовком символ "перечень". Поэтому придется для данного случая создать еще один прием:

$\forall_{ab}(\text{наложение}((a),b,(1))=\text{префикс}(a,b))$

При вводе одноэлементных наборов (a),(1) могут возникнуть трудности: формульный редактор не поймет, что скобки означают одноэлементные наборы, и проигнорирует их. Потому либо нужно будет использовать специальную клавишу Ctr-н, либо набрать прием текстовым редактором.

После того, как пакет создан, нужно еще ввести прием, обращающийся к нему:

$\forall_{abcd}(a=\text{наложение}(b,c,d) \to \text{наложение}(b,c,d)=a)$

Прием имеет заголовок "второйтерм". Антецедент выделен указателем "идентификатор", причем правая часть антецедента обрабатывается нормализатором "нормналожение". Фильтр "не(заголовок(x1 наложение))" контролирует невырожденность обращения.

6. Вводим задачу на описание с условием "Вставка$((1,2,3,4),x,y)=(1,5,2,3,4))$"; x,y - неизвестные, причем требуется найти все возможные их значения. Теорема приема должна сравнивать два набора и находить такой номер позиции k, до которой наборы совпадают, а после - элементы первого набора совпадают с элементами второго, имеющими на единицу большие номера. Если при этом на k - й позиции элементы различны, то место вставки определяется однозначно. Таким образом, приходим к следующему приему:

   $\forall_{abkmnxy}(m=n+1\ \&\ k \in \{1,\dots,n\}\ \&\ \neg(a(k)=b(k))\ \&\ \forall_i(i \in \{1,\dots,k-1\} \to a(i)=b(i))\ \&\ \forall_i(i \in \{k,\dots,n\} \to a(i)=b(i+1)) \to (\text{Вставка}(\lambda_i(a(i), i \in \{1,\dots,n\}),x,y)=\lambda_j(b(j), j \in \{1,\dots,m\})) \leftrightarrow (x=k\ \&\ y=b(k)))$

   Прежде всего, идентифицируются наборы a, b. Соответствующие им описатели "отображение" в левой части эквивалентсноти выделены указателями "развертка". Сами переменные a,b выделены указателем "отображение". После идентификации наборов определяются натуральные константы m,n. Первый антецедент, выделенный указателем "программа", проверяет, что второй набор на единицу длиннее первого. Второй антецедент тоже выделен указателем "программа" - он перечисляет номера k сравниваемых позиций. Третий антецедент выделен указателем "легковидеть". Так как тип элементов наборов априори неизвестен, для установления различия элементов используем вспомогательную задач на доказательство. Четвертый и пятый антецеденты обеспечивают циклы сравнения наборов до и после позиции k. Они выделены указателем "идентификатор". Прием имеет заголовок "второйтерм". Фильтры возьмем, например, следующие: "уровень(2)", "тип(описать)", "условие".

7. Вводим условие задачи на преобразование "кортежпар$((1,2,3),(4,5,6))$". Очевидно, необходима следующая теорема приема:

   $\forall_{abn}(\text{кортежпар}(\lambda_i(a(i), i \in \{1,\ldots,n\}), \lambda_j(b(j), j \in \{1,\ldots,n\})) = \lambda_i((a(i),b(i)), i \in \{1,\ldots,n\}))$

   Указатель "развертка(фикс(0 1 1)фикс(0 1 2)фикс(0 2))" определяет работу с конечными наборами. Переменные a, b выделены указателем "отображение".

8. Вводим задачу на описание с условиями "последовательность(a,N)", "подпоследовательность$(\lambda_i(i, i-\text{натуральное}), a)$", "подпоследовательность$(\lambda_i(i^2, i-\text{натуральное}), a)$". В ней нужно найти пример значения неизвестной a. Создадим прием, дающий параметрическое описание для последовательности, имеющей заданную подпоследовательность. Простейший способ - разместить элементы этой подпоследовательности на четных позициях, а на нечетных - элементы новой последовательности, играющей роль параметра. Теорема приема имеет вид:

   $\forall_{xaA}(\exists_c(\text{последовательность}(c, A)\ \&\ x = \lambda_j((a(j/2)\ \text{при}\ j-\text{even}, \text{иначе}\ c((j+1)/2)), j-\text{натуральное})) \to \text{последовательность}(x, A)\ \&\ \text{подпоследовательность}(\lambda_i(a(i), i-\text{натуральное}), x))$

   Заголовок приема - "подборзначений"; он реализует попытку неэквивалентной замены условий. Указатель "подборзначений(1)" означает, что роль заменяющего утверждения играет антецедент. Указатель "комментарий(1 серия)" передает комментарий "серия"к заменяющему утверждению во вспомогательную задачу. Таким образом, это утверждение будет рассматриваться как параметрическое описание с параметром c. Функциональная переменная a выделена указателем "отображение". Прием имеет фильтры "уровень(4)", "тип(описать)", "цель(пример)", "неизвестная(x23)", "известно(фикс(0 2 1))", "контекст(новоеусловие(x2)не(равно(x2 фикс(0 1)))входит(x23 x2))". Последний из фильтров означает, что имеется еще какое-то условие на неизвестную последовательность. Иначе можно было бы просто отождествить ее с известной подпоследовательностью.

   После срабатывания введенного приема и исключения неизвестной a, явно выраженной через новую неизвестную последовательность, возникнет задача, имеющая условие:

   $\text{подпоследовательность}(\lambda_i(i^2, i-\text{натуральное}), \lambda_n((n/2\ \text{при}\ n-\text{even}, \text{иначе}\ b((n+1)/2)), n-\text{натуральное}))$.

   Здесь b - новая неизвестная последовательность. Нам нужен еще один прием, который будет связывать указанную подпоследовательность элементов i^2 с b. Этот переход неэквивалентный, так что прием снова имеет заголовок "подборзнчений". Его теорема имеет вид:

$\forall_{cpqA}$(последовательность(q, A) & подпоследовательность(c, q) $\rightarrow$ подпоследовательность$(c, \lambda_n((p(n)$ при $n -$ even, иначе $q((n+1)/2)), n -$ натуральное$)))$

Указатель "подборзначений(2)" выделяет заменяющий антецедент. Прочие указатели и фильтры - стандартные. Хотя данный прием и может показаться слишком искусственным, он всего лишь учитывает специальный вид параметрического задания неизвестных последоватеьностей, определяемый предыдущим приемом.

Для завершения решения задачи нам понадобится еще один прием - отождествление неизвестной последовательности x с известной подпоследовательностью, если прочих условий на x нет. Теорема приема такова:

$\forall_{axA}$(последовательность(a, A) & $x = a \rightarrow$ последовательность(x, A) & подпоследоватеьность$(a, x))$

Заголовок приема - "подборзначений". Указатели "блокпроверок(1)", "подборзначений(2)" задают обработку антецедентов. Фильтр "не(контекст(новоеусловие(x2)не(равно(x2 фикс(0 1)))входит(x23 x2)))" обеспечивает отсутствие прочих условий с неизвестной x. Оставшиеся фильтры - "уровень(2)", "тип(описать)", "цель(пример)", "неизвестная(x23)", "известно(x1)".

9. Вводим задачу на упрощение выражения "индикатор$(\{1,2,3\},\{1,3\},1)$". Переход к табличному заданию такого выражения будем считать (по крайней мере, на момент рассмотрения данной задачи) преобразованием общей стандартизации. Возможно, впоследствии возникнут какие-то разумные ограничения этого перехода. Теорема приема имеет вид:

   $\forall_{aAnb}$(индикатор$(\{;\lambda_i(a(i), i \in \{1,\ldots,n\})\}, A, b) =$ таблица$\{;\lambda_i(a(i) \rightarrow (b$ при $a(i) \in A,$ иначе $0), i \in \{1,\ldots,n\})\})$

   Указатель "развертка(фикс(0 1 1 1)фикс(0 2 1 1))" определяет идентификацию и формирование описателей "отображение" как конечных наборов. Условное выражение в заменяющем терме обрабатывается нормализатором "нормвариант", а его подутверждение $a(i) \in A$ - нормализатором "нормпринадлежит".

10. Вводим задачу на преобразование выражения "функстепень$(\lambda_i(i^2, i -$натуральное$), 3)$". Заметим, что "функстепень" вводится и прорисовывается формульным редактором как обычная степень, причем перед началом решения задачи происходит автоматическое преобразование ее в "функстепень". Для рекурсивного определения n - кратного последовательного применения функции к заданному аргументу создаем следующие два приема:

    $\forall_{fgABn}(\forall_x(A(x) \rightarrow A(f(x)))$ & $(\lambda_x(f(x), A(x)))^{n-1} = \lambda_x(g(x), B(x)) \rightarrow (\lambda_x(f(x), A(x)))^n = \lambda_x(f(g(x)), A(x)))$

    $\forall_{fA}((\lambda_x(f(x), A(x)))^1 = \lambda_x(f(x), A(x)))$

    Здесь тоже "функстепень" вводится как обычная степень, а при компиляции приема модифицируется. Первый антецедент первой теоремы приема выделен указателем "следствие". Он проверяет, что область значений рассматриваемой функции содержится в ее области определения, и таким образом решает проблему согласования областей определения при последовательных отображениях. Второй антецедент выделен указателем "идентификатор", а левая часть его обрабатывается вспомогательной задачей на упрощение. Здесь и реализуется шаг рекурсии. Фильтр "натуральное(x14)" ограничивает попытки применения приема случаями, когда величина степени - натуральная константа. Рекомендуем самостоятельно реализовать эту же схему рекурсии в виде приемов нормализатора "нормфункстепень" и добавить прием, обращающийся к данному нормализатору. Нормализатор "нормфункстепень" уже имеет приемы, предназначенные для специальных случаев, так что новые приемы следует вводить на более высоких уровнях, чтобы они не мешали срабатыванию уже введенных.

Глава 6

Приемы, связанные с мощностями множеств и комбинаторикой

Приемы, связанные с мощностью множеств, находятся в подразделе "мощности" раздела "Теория множеств" оглавления базы приемов. В основном, они были созданы при рассмотрении простейших комбинаторных задач (см. подраздел задачника "Комбинаторика", расположенный в разделе "Теория множеств"). Как правило, такая задача имела текстовую формулировку, решателю же предлагалась некоторая версия ее логической формализации. Для удобства последняя сопровождалась оригинальным текстом, который можно просмотреть, нажав клавишу "н". Если нужно одновременно просматривать и текст, и логическую формализацию задачи, следует прокруткой переместить верхнюю отделяющую задачу горизонтальную линию в верхнюю часть экрана и нажать клавишу "$+$". Переход от текстовой постановки комбинаторной задачи к ее логической формализации в будущем тоже предполагается автоматизировать, и сохраняемые в задачнике примеры предоставят необходимый для этого обучающий материал.

Собственно к разделу "мощности" отнесено сравнительно мало логических символов, так как многие встречающиеся в комбинаторных задачах понятия, связанные с множествами и функциями, уже введены в предыдущих разделах. Перечислим понятия данного раздела. Выражение "мощность(A)" обозначает мощность множества A. Формульный редактор прорисовывает символ "мощность" как card. Мощность множества - кардинальное число; в случае конечных множеств - целое неотрицательное число. Таким образом, утверждение "число(мощность(A))" истинно тогда и только тогда, когда множество A - конечное. Утверждение "конечное(A)" означает, что множество A конечное. Константы "счетное", "континуум" обозначают кардинальные числа - мощности счетного и континуального множеств. Выражение "сочетания(A n)" обозначает множество всех подмножеств множества A, имеющих мощность n. Впрочем, символ "сочетания" в задачах появляется крайне редко - обычно используется его явное определение через описатель "класс". Если A - конечное семейство (функция, определенная на конечном множестве), то посредством "комплект(A)" обозначается набор кратностей элементов семейства, упорядоченных по невозрастанию. Утверждение "конечнпересек(A B)" означает, что пересечение множеств A, B конечно.

6.1 Примеры логической формализации комбинаторных задач

Чтобы проиллюстрировать типичные конструкции с функциями и множествами, возникающие при логическом представлении комбинаторных задач, рассмотрим несколько примеров. Это прояснит ту роль, которую играют представленные в разделе приемы.

1. Начнем с задачи, имеющей следующее условие: "Экскурсанты разделились на две равные группы для розыска заблудившегося товарища. Среди них есть только 4 человека, знакомые с местностью. Каким числом способов они могут разделиться так, чтобы в каждую группу вошло два человека, знающих местность, если всего их шестнадцать человек ?". Задачу на подсчет числа способов будем оформлять как задачу на преобразование, условием которой является выражение $\mathrm{card}(\ldots)$ для мощности множества этих способов. Начнем с перечисления посылок задачи. Обозначим через A множество всех экскурсантов, а через D - подмножество экскурсантов, знакомых с местностью. Тогда имеем три посылки: $\mathrm{card}A = 16$, $\mathrm{card}D = 4$, $D \subseteq A$. Обозначим через B, C группы, на которые должны разделиться экскурсанты. Возможны два понимания условия задачи - либо эти группы различаются: одна считается первой, а другая - второй, либо безразлично, какая из них первая, а какая вторая. В первом случае нужно подсчитать число упорядоченных пар (B, C), удовлетворяющих условиям задачи, во втором случае - число неупорядоченных пар, т.е. двухэлементных множеств $\{B, C\}$. В обоих случаях условия на B, C одни и те же, так что сначала выпишем их: $B - \mathrm{set}$, $C - \mathrm{set}$, $A = B \cup C$, $B \cap C = \emptyset$, $\mathrm{card}B = \mathrm{card}C$, $\mathrm{card}(B \cap D) = 2$, $\mathrm{card}(C \cap D) = 2$. Если требуется найти число упорядоченных пар, то условием задачи служит выражение $\mathrm{card}(\mathrm{set}_{BC} P(B, C))$, где $P(B, C)$ - конъюнкция приведенных выше условий на множества B, C. Если же требуется найти число неупорядоченных пар, то условие задачи приобретает вид: $\mathrm{card}(\mathrm{set}_x(\exists_{BC}(x = \{B, C\} \ \& \ P(B, C))))$. Целевая установка задачи - "упростить в о.д.з".

2. Во многих комбинаторных задачах приходится подсчитывать число отображений заданного вида. Приведем следующий пример: "Из группы в 12 человек ежедневно в течение 6 дней выбирают двух дежурных. Определить количество различных списков дежурных, если каждый человек дежурит один раз." Список дежурных представляет собой функцию, определенную на множестве номеров человек (от 1 до 12) и сопоставляющую каждому человеку номер дня, когда он дежурит (от 1 до 6). Дополнительным ограничением является то, что каждый номер дня должен являться значением в двух различных точках. Таким образом, условием задачи имеет вид:

   $\mathrm{card}(\mathrm{set}_f(\text{Отображение}(f, \{1, \ldots, 12\}, \{1, \ldots, 6\}) \ \& \ \forall_i (i \in \{1, \ldots, 6\} \to \mathrm{card}(\text{слой}\ (f, i)) = 2)))$.

3. Еще один пример, когда задача сводится к подсчету числа отображений. "Из 10 теннисисток и 6 теннисистов составляют 4 смешанные пары. Сколькими способами это можно сделать ?". Состав смешанных пар определится, если выбрать, например, четырехэлементное подмножество в множестве 10 теннисисток и определить на нем функцию, сопоставляющую второй элемент пары.

Эта функция должна принимать в различных точках различные значения, т.е. быть биективной (по терминологии решателя - взаимно-однозначной). Таким образом, получаем следующее условие задачи:

$\mathrm{card}(\mathrm{set}_f(\exists_A(A \subseteq \{1, \ldots, 10\}$ & $\mathrm{card} A = 4$ & Отображение$(f, A, \{1, \ldots, 6\})$ & взаимнооднозначно$(f))))$

4. В заключение приведем пример, где возникает понятие разбиения. "В турнире участвуют 16 шахматистов. Определить количество различных расписаний первого тура (расписания считаются различными, если отличаются участниками хотя бы одной партии; цвет фигур и номер доски не учитывается". Очевидно, что каждое расписание соответствует некоторому разбиению множества шахматистов на двухэлементные подмножества. Таким образом, условие задачи имеет следующий вид:

   $\mathrm{card}(\mathrm{set}_x($разбиение$(\{1, \ldots, 16\}, x)$ & $\forall_y(y \in x \to \mathrm{card} y = 2)))$.

6.2 Непосредственное определение мощности

Комбинаторные задачи обычно сводятся, после ряда промежуточных переходов, к нахождению мощностей некоторых стандартных множеств. Последнее выполняется приводимыми ниже приемами.

Мощность пустого множества

Мощность пустого множества заменяется на 0:

$\mathrm{card}\emptyset = 0$

Мощность одноэлементного множества

Мощность одноэлементного множества заменяется на 1:

$\forall_a(\mathrm{card}\{a\} = 1)$

Мощность не более чем одноэлементного класса

$\forall_{ab}(\mathrm{card}(\mathrm{set}_x(x = a\ \&\ b)) = (1$ при $b,$ иначе $0))$

Мощность пары

Хотя ниже будет рассмотрен случай произвольных перечней, для двухэлементного перечня введен особый прием:

$\forall_{ab}(\mathrm{card}\{a, b\} = (1$ при $a = b,$ иначе $2))$

Кроме того, введен прием, определяющий мощность подмножества пары, выделенного заданным условием:

$\forall_{abP}(\mathrm{card}(\mathrm{set}_x(x \in \{a, b\}\ \&\ P(x))) = (2$ при $P(a)\ \&\ P(b),$ иначе $(0$ при $\neg(P(a))\ \&\ \neg(P(b)),$ иначе $1)))$

Мощность перечня

Простейший случай - переход к меньшему перечню, если удается усмотреть отличие некоторого элемента от всех остальных элементов:

$\forall_{ab}(\neg(a \in \{;b\}) \rightarrow \mathrm{card}\{a;b\} = \mathrm{card}\{;b\} + 1)$

Здесь антецедент выделен указателем "блокпроверок".

Если мощность перечня встречается в задаче на описание, причем некоторый элемент перечня содержит неизвестные, то реализуется разбор случаев по принадлежности этого элемента остатку перечня. Теорема приема имеет вид: $\forall_{ab}(a \in \{;b\} \vee \neg(a \in \{;b\}))$. Указатель "контрольвывода(мощность(перечень (префикс(x1 x2))))" задает инициализацию применения приема при усмотрении выражения "мощность(перечень(префикс(x1 x2)))". Прием имеет заголовок "выводусловия" и фильтры "уровень(5)", "условие", "тип(описать)", "не(известно(x1))". Указатель "примечание(разборслучаев)" дает установку на немедленный разбор случаев по выведенной дизъюнкции. Указатель "список(префикс(x1 x2))" определяет идентификацию элемента a с произвольным элементом перечня.

Наконец, в случае задачи на преобразование для отбрасывания элемента перечня используется условное выражение:

$\forall_{abp}(p = (a \in \{;b\}) \rightarrow \mathrm{card}\{a;b\} = \mathrm{card}\{;b\} + (0\text{при } p, \text{иначе } 1))$

Антецедент выделен указателем "идентификатор". Правая часть его обрабатывается вспомогательной задачей на описание, причем неизвестной служит некоторый параметр x3 выражения a, идентифицируемый через указатель "контекст(входит(x3 параметры(x1)))". Переменной p присваивается результат разрешения условия принадлежности элемента a остатку перечня относительно x3.

Мощность множества целых чисел, заключенных между двумя целыми числами

В зависимости от того, каким образом задано множество, применяется один из следующих приемов:

$\forall_{nma}(n - \text{целое} \;\&\; m - \text{целое} \;\&\; a = m - n \;\&\; 0 \leq a \rightarrow \mathrm{card}(\mathrm{set}_k(k - \text{целое} \;\&\; n \leq k \;\&\; k \leq m)) = a + 1)$

$\forall_{ij}(0 \leq j - i + 1 \rightarrow \mathrm{card}(\{i, \ldots, j\}) = j - i + 1)$

$\forall_{mn}(0 \leq m + n \;\&\; m - \text{целое} \;\&\; n - \text{целое} \rightarrow \mathrm{card}(\mathrm{set}_j(j - \text{целое} \;\&\; 0 \leq m - j \;\&\; 0 \leq n + j)) = m + n + 1)$

Во всех этих случаях предварительно проверяется непустота множества. Если такая проверка не удается, применяется условное выражение:

$\forall_{ij}(i - \text{целое} \;\&\; j - \text{целое} \rightarrow \mathrm{card}(\{i, \ldots, j\}) = (j - i + 1\text{при}0 \leq j - i, \text{иначе}0))$

Мощность множества целых чисел, заключенных между двумя целыми числами и имеющих заданное значение по модулю третьего числа

$\forall_{abcn}(n - \text{натуральное} \;\&\; a - \text{целое} \;\&\; b - \text{целое} \;\&\; 0 \leq b - a \;\&\; c \in \{0, \ldots, n-1\} \rightarrow \mathrm{card}(\mathrm{set}_x(x \in \{a, \ldots, b\} \;\&\; x(\mathrm{mod}\; n) = c)) = [(b - a - (c - a)(\mathrm{mod}\; n))/n] + 1)$

Мощность множества целых чисел на промежутке

Если нужно найти число точек одномерной решетки, расположенных между двумя вещественными числами, применяется следующий прием:

$\forall_{abcpqr}(r = [(aq+b)/c] + [-(ap+b)/c] + 1 \to \mathrm{card}(\mathrm{set}_x(p \le x \ \& \ x \le q \ \& \ (ax+b)/c -$ целое$)) = \max(r,0))$

Указатель "заменазнака(минус x1 x3)" разрешает идентификацию с отрицательными a, c. Первый антецедент выделен указателем "идентификатор". Подвыражения его правой части обрабатываются нормализаторами общей стандартизации.

Мощность множества подмножеств конечного множества

$$\forall_a(\text{конечное}(a) \to \mathrm{card}(\mathrm{set}_x(x \subseteq a \ \& \ x - \mathrm{set})) = 2^{\mathrm{card}(a)})$$

Первый антецедент обрабатывается проверочным оператором "усмконечное", распознающим конечные множества.

Мощность множества отображений одного конечного множества в другое

$\forall_{AB}(\text{конечное}(A) \ \& \ \text{конечное}(B) \to \mathrm{card}(\mathrm{set}_f(\text{Отображение}(f, A, B))) = (\mathrm{card}B)^{\mathrm{card}A})$

Число подмножеств заданной мощности

Для определения числа сочетаний введены следующие два приема:

$\forall_{an}(n -$ целое $\& \ 0 \le n \ \& \ \text{конечное}(a) \to \mathrm{card}(\mathrm{set}_x(x - \mathrm{set} \ \& \ x \subseteq a \ \& \ \mathrm{card}(x) = n)) = C^{\ n}_{\mathrm{card}(a)})$

$\forall_{Amn}(\mathrm{card}(A) = n \ \& \ n -$ число $\to \mathrm{card}(\text{сочетания}(A, m)) = C^m_n)$

В первом приеме первые два антецедента обрабатываются вспомогательной задачей на доказательство. Во втором приеме первый антецедент выделен указателем "идентификатор", а левая его часть обрабатывается нормализатором "норммощность". Если не удается установить неотрицательность n, то первый прием модифицируется с использованием условного выражения:

$\forall_{an}(n -$ целое $\& \ \text{конечное}(a) \to \mathrm{card}(\mathrm{set}_x(x - \mathrm{set} \ \& \ x \subseteq a \ \& \ \mathrm{card}(x) = n)) = (C^n_{\mathrm{card}(a)}$ при $0 \le n,$ иначе $0))$

Здесь первые два антецедента выделены указателями "блокпроверок".

Число отображений с заданными мощностями прообразов

Чтобы подсчитать число таких отображений множества A в множество B, у которых для каждого элемента B задано количество точек, переводимых в этот элемент, используется следующий прием:

$$\forall_{ABg}(\text{конечное}(A) \ \& \ \text{конечное}(B) \to \mathrm{card}(\mathrm{set}_f(\text{Отображение}(f, A, B) \ \& \ \forall_y(y \in B$$

$$\to \mathrm{card}(\text{слой}(f, y)) = g(y)))) = \mathrm{card}(A)!/\prod_{y, y \in B}(g(y)!))$$

Мощность множества взаимно-однозначных отображений одного конечного множества в другое

Используется следующий прием:

$\forall_{ABmn}(m = \text{card}(B)\ \&\ m - \text{целое}\ \&\ n = m - \text{card}(A)\ \&\ 0 \le n \to \text{card}(\text{set}_f(\text{Отображение}(f, A, B)\ \&\ \text{взаимнооднозначно}(f))) = m!/n!)$

Если множества A, B зависят от некоторых параметров x, однако мощности их неизменны, то для подсчета числа комбинаций взаимно-однозначного отображения f множества $A(x)$ в $B(x)$ и значений x, удовлетворяющих также дополнительным условиям $P(x)$, предыдущий прием модифицируется:

$\forall_{ABPmn}(m = \text{card}(B(x))\ \&\ m - \text{целое}\ \&\ n = m - \text{card}(A(x))\ \&\ 0 \le n \to \text{card}(\text{set}_{fx}(\text{Отображение}(f, A(x), B(x))\ \&\ \text{взаимнооднозначно}(f)\ \&\ P(x))) = (m!\,\text{card}(\text{set}_x(P(x)))\ /n!))$

Чтобы гарантировать независимость мощностей множеств $A(x)$, $B(x)$ от x, введены фильтры "не(пересекаются(x23 x13))", "не(пересекаются(x23 терм(мощность(значение(x26 x23)))))". Оба выражения для мощности обрабатываются вспомогательной задачей на упрощение. Фильтр "не(равно(x23 логсимвол(пустоеслово)))" отсекает случай пустого набора параметров x, уже рассмотренный в первом приеме.

Число разбиений конечного множества на подмножества одинаковой мощности

Введены два приема - для различного представления множества разбиений:

$\forall_{mna}(n = \text{card}(a)/m\ \&\ n - \text{целое}\ \&\ m - \text{натуральное} \to \text{card}(\text{set}_x(\text{разбиение}(a, x)\ \&\ x - \text{set}\ \&\ \forall_y(y \in x \to \text{card}(y) = m))) = (\text{card}(a))!/(m!)^n \cdot n!)$

$\forall_{mnA}(\text{card}(A) - mn = 0 \to \text{card}(\text{set}_x(\text{кортеж}(x, n, \text{сочетания}(A.m))\ \&\ \text{разбиение}(A, \{; x\}))) = (mn)!/(m!)^n)$

Во втором приеме антецедент выделен указателем "идентификатор", причем выражение card(A) сначала обрабатывается нормализатором "норммощность", а затем упрощается вспомогательной задачей на преобразование.

Число разбиений конечного множества, если заданы мощности классов разбиения и количества этих классов

$\forall_{ABf}(\text{конечное}(A) \to \text{card}(\text{set}_x(\text{разбиение}(A, x)\ \&\ x - \text{set}\ \&\ \text{set}_y(\exists_z(z \in x\ \&\ \text{card}(z) = y)) = B\ \&\ \forall_y(y \in B \to \text{card}(\text{set}_z(z \in x\ \&\ \text{card}(z) = y)) = f(y)))) = (\text{card}(A))!/\prod_{y, y \in B}((y!)^{f(y)} \cdot f(y)!))$

Число всех разбиений конечного множества

В этом приеме используются числа Белла, обозначаемые "числобелла(n)". Для их вычисления в подразделе "Комбинаторные функции" раздела "Элементарная алгебра" создан нормализатор "нормчислобелла".

$\forall_A(\text{конечное}(A) \to \text{card}(\text{set}_x(\text{разбиение}(A, x)\ \&\ x - \text{set})) = \text{числобелла}(\text{card}(A)))$

Добавлен прием, определяющий число неодноэлементных разбиений:

$\forall_A(\text{конечное}(A) \to \text{card}(\text{set}_x(\text{разбиение}(A, x)\ \&\ x - \text{set}\ \&\ 1 < \text{card}(x))) = \text{числобелла}(\text{card}(A)) - 1)$

Число сочетаний с повторениями

Сочетание с повторениями представляется набором целых неотрицательных чисел, указывающих число экземпляров, в которых берется соответствующий элемент. В зависимости от того, каким образом задано множество сочетаний с повторениями, используется один из следующих приемов:

$\forall_{mn}(n$ – натуральное & m – целое & $0 \leq m \to \mathrm{card}(\mathrm{set}_f($кортеж$(f, n, \mathbf{N}+)$ & $\sum_{i=1}^{n} f(i) - m = 0)) = C_{n+m-1}^{m})$

$\forall_{mn}(m$ – целое & $0 \leq m \to \mathrm{card}(\mathrm{set}_x(\forall_i(i \in \{1, \ldots, n\} \to x(i)$ – целое & $0 \leq x(i))$ & $\sum_{i=1}^{n} x(i) - m = 0)) = C_{n+m-1}^{m})$

$\forall_{mn}(m$ – целое & $0 \leq m \to \mathrm{card}(\mathrm{set}_x(\forall_i(i \in \{1, \ldots, n\} \to x(i)$ – натуральное) & $\sum_{i=1}^{n} x(i) - m = 0)) = C_{m-1}^{n-1})$

$\forall_{mn}(n$ – натуральное & m – целое & $0 \leq m \to \mathrm{card}(\mathrm{set}_f($кортеж$(f, n, N)$ & $\sum_{i=1}^{n} f(i) - m = 0)) = C_{m-1}^{n-1})$

Второй и третий приемы относятся к наборам фиксированной длины n, элементы которых образуют связывающую приставку x описателя "класс". Поэтому они имеют указатели "кортежпеременных(x23)", "контекст(равно(x14 длинанабора(x23)))", "развертка(фикс(0 1 1 2 1)фикс(0 1 1 2 2 1 1))". Последний указатель определяет идентификацию кванторной импликации по i с конъюнкцией утверждений, а также идентификацию конечной суммы от 1 до n с обычной суммой n слагаемых.

Число перестановок

Для нахождения числа перестановок создан следующий прием:

$\forall_A(\mathrm{card}(A) = a$ & a – число $\to \mathrm{card}(\mathrm{set}_x($перестановка$(x, A))) = a!)$

Отдельно рассмотрены случаи, когда нужно найти число перестановок с дополнительным условием на элемент, располагающийся в заданной позиции либо на элементы двух различных заданных позиций:

$\forall_{AinP}(\mathrm{card}(A) = n$ & $i \in \{1, \ldots, n\} \to \mathrm{card}(\mathrm{set}_f($перестановка$(f, A)$ & $P(f(i))$ & f – функция$)) = \mathrm{card}(\mathrm{set}_x(x \in A$ & $P(x))) \cdot (n-1)!)$

$\forall_{AijnP}(\mathrm{card}(A) = n$ & $i \in \{1, \ldots, n\}$ & $j \in \{1, \ldots, n\}$ & $\neg(i - j = 0) \to \mathrm{card}(\mathrm{set}_f($перестановка$(f, A)$ & f – функция & $P(f(i), f(j))))) = \mathrm{card}(\mathrm{set}_{xy}(x \in A$ & $y \in A$ & $P(x, y))) \cdot (n-2)!)$

Указатель "контекст(позиция(x1 фикс(0 1 1 2))вид(x1 значение(x6 x9)))" определяет идентификацию номера позиции i по вхождению выражения $f(i)$ в условие на перечисляемые перестановки f. В последнем приеме аналогичный указатель введен для номера j. Указатель "новаргумент(x40 x6 фикс)" обеспечивает проверку того, что дополнительные условия на f относятся только к значениям в точках i, j. Мощность множества A вычисляется в первом антецеденте (он выделен указателем "идентификатор") с помощью вспомогательной задачи на преобразование.

Использование посылки

Если в контексте уже имеется равенство, дающее значение мощности рассматриваемого множества, то оно используется для исключения символа "мощность":

$\forall_{ab}(\mathrm{card}(a) = b \to \mathrm{card}(a) = b)$

Прием имеет фильтр "не(входит(мощность x2))", проверяющий отсутствие символа "мощность" в заменяющем выражении b, а также фильтр "не(контекст(подтерм(и(равно(теквхожд x5)равно(мощность(x1)x3)x4))единица(истина x4)))", блокирующий замену в конъюнкции с несколькими равенствами для данной мощности.

Мощность множества упорядоченных пар

Число упорядоченных пар элементов заданного конечного множества определяется следующим приемом:

$\forall_A(\text{конечное}(A) \to \text{card}(\text{set}_{ij}(i \in A \ \& \ j \in A \setminus \{i\})) = \text{card}(A)(\text{card}(A) - 1))$

Указатель "элемент(x9)" разрешает идентифицировать переменную i с произвольным (не обязательно первым) элементом связывающей приставки. Это избавляет от необходимости создавать дополнительный прием с переставленными переменными. Если нужно рассматривать пары, у которых первый элемент принадлежит заданному множеству, а второй - его заданному надмножеству, то работает другой прием:

$\forall_{ABC}(\text{конечное}(B) \ \& \ A \subseteq B \setminus C \to \text{card}(\text{set}_{ij}(i \in A \ \& \ j \in B \setminus (C \cup \{i\}))) =$
$\text{card}(A)(\text{card}(B \setminus C) - 1))$

Еще один прием - для нахождения числа упорядоченных по возрастанию пар вещественных чисел. Фактически, это уже подсчет числа неупорядоченных пар:

$\forall_A(\text{конечное}(A) \ \&A \subseteq \mathbb{R} \to \text{card}(\text{set}_{ij}(i < j \ \& \ i \in A \ \& \ j \in A)) = C^2_{\text{card}(A)})$

Подмножества, имеющие не более чем одноэлементные пересечения с классами разбиения на равномощные подмножества

Пусть задано разбиение B множества A на n равномощных подмножеств, каждое из которых имеет p элементов. Если требуется перечислить m - элементные подмножества множества A, имеющие в каждом классе разбиения не более одного представителя, то нужно рассмотреть всевозможные m - элементные подмножества классов разбиения, и в каждом из них выбрать по одному элементу. Это дает $C_n^m p^m$ вариантов. Таким образом, получаем следующую теорему приема:

$\forall_{ABmnp}(\text{разбиение}(A, B) \ \& \ \text{card}(B) = n \ \& \ \forall_i(i \in B \to \text{card}(i) = p) \ \& \ n - \text{число} \ p -$ $\text{число} \to \text{card}(\text{set}_x(\forall_y(y \in B \to \text{card}(x \cap y) \leq 1) \ \& \ x \subseteq A \ \& \ x - \text{set} \ \& \ \text{card}(x) = m)) =$ $(0 \text{ при } n < m, \text{ иначе } C_n^m p^m))$

Второй антецедент выделен указателем "идентификатор". Здесь нормализатор "норммощность" предпринимает попытку вычислить число классов эквивалентности. Четвертый и пятый антецеденты выделены указателями "блокпроверок". Они проверяют конечность числа классов и каждого класса.

Двоичный куб

Создано несколько простых приемов, позволяющих подсчитывать мощности подмножеств двоичного n - мерного куба (точками этого куба являются наборы из нулей и единиц, имеющие длину n).

1. Число точек куба.

   $\forall_n(n - \text{натуральное} \to \text{card}(\text{двкуб}(n) = 2^n)$

2. Число точек $n-1$ - мерной грани.

   $\forall_{nia}(n -$ натуральное & $i \in \{1, \dots, n\}$ & $a -$ boolean $\to$ card(set$_x$(двнабор(x,n) & $x(i) = a)) = 2^{n-1})$

3. Число наборов, имеющих заданное число нулей либо единиц.

   $\forall_{amn}(0 \leq m$ & $m -$ целое & $0 \leq n - m$ & $n -$ натуральное & $a -$ boolean $\to$ card(set$_x$(двнабор(x,n) & колич$(x,a) = m)) = C_n^m)$

   Здесь "колич(x,a)" - число разрядов набора x, равных a.

4. Число наборов, имеющих не менее (не более) заданного количества нулей либо единиц.

   $\forall_{mn}(n-$натуральное & $m-$целое & $0 \leq m$ & $0 \leq n-m \to$ card(set$_x$(двнабор & $m \leq$ колич$(x,a))) = \sum_{i=m}^{n} C_n^i)$

   $\forall_{mn}(n-$натуральное & $m-$целое & $0 \leq m$ & $0 \leq n-m \to$ card(set$_x$(двнабор(x,n) & колич$(x,a) \leq m)) = \sum_{i=0}^{m} C_n^i)$

   Приемы применяются для конкретных численных значений m, n, причем указатели "развертка" определяют запись конечной суммы в виде обычной суммы нескольких слагаемых.

Бесконечные множества

Для простейших случаев усмотрения счетного либо континуального множества введены следующие приемы:

1. Множества целых либо натуральных чисел

   card$(\mathbb{Z})$ = счетное

   card$(\mathbb{N})$ = счетное

2. Множество рациональных чисел

   card$(\mathbb{Q})$ = счетное

3. Множество рациональных точек промежутка

   $\forall_{abcd}(0 < b - a \to$ card(set$_x$$(x -$ rational & $x \in [a,b]))$ = счетное)

   Здесь формульная запись $[a,b]$ соответствует промежутку произвольного типа (в скобочной записи добавляются переменные c,d, уточняющие тип концов).

4. Множество вещественных чисел

   card$(\mathbb{R})$ = континуум

5. Числовой промежуток

   $\forall_{ab}(\text{card}([a,b]) = 1 \leftrightarrow a = b)$

   Промежуток $[a,b]$ берется замкнутый.

   $\forall_{abcd}(0 < b - a \to$ card$([a,b])$ = континуум)

   Здесь уже тип промежутка произвольный.

Для случая, когда в множестве выделяется промежуток, введены несколько приемов, отличающихся способом установления того, что мощность остальной части не более чем континуальна:

$\forall_{abcdA}(0 < b - a \ \&$ конечное$(A) \to \mathrm{card}(A \cup [a,b]) =$ континуум$)$

$\forall_{abcdA}(0 < b - a \ \& \ (\mathrm{card}(A) =$ континуум$\vee\mathrm{card}(A) =$ счетное$) \to \mathrm{card}(A \cup [a,b]) =$ континуум$)$

Для определения мощности множества A прием использует нормализатор "норммощность".

$\forall_{abcdA}(0 < b - a \ \& \ A \subseteq \mathbb{R} \to \mathrm{card}(A \cup [a,b]) =$ континуум$)$

6. Мощность множества пар чисел, удовлетворяющих линейному уравнению.

   $\forall_{abP}((\neg(a = 0) \vee \neg(b = 0)) \ \&$ конечное$(P) \to \mathrm{card}(P \cup \mathrm{set}_{xy}(ax + by + c = 0 \ \& \ x -$ число $\& \ y -$ число$)) =$ континуум$)$

   Указатели "подстановка(фикс(0 1 1 2 3 1 1 1)x1 0)", "подстановка(фикс(0 1 1 2 3 1 1 2)x2 0)" учитывают случаи, когда в линейном соотношении коэффициент при x либо y оказывается равен 0.

7. Мощность бесконечной серии.

   $\forall_{ab}(\mathrm{card}(\mathrm{set}_x(\exists_n(n -$ целое $\& \ x = an + b))) = (1$ при $a = 0,$ иначе счетное$))$

8. Мощность множества целых чисел, больших либо меньших заданного числа.

   $\forall_a(a -$ число $\to \mathrm{card}(\mathrm{set}_x(x -$ целое $\& \ a \leq x)) =$ счетное$)$

   $\forall_a(a -$ число $\to \mathrm{card}(\mathrm{set}_x(x -$ целое $\& \ x \leq a)) =$ счетное$)$

Число наборов элементов конечного множества, имеющих заданную длину

$\forall_{nA}($конечное$(A) \ \& \ n -$ целое $\& \ 0 \leq n \to \mathrm{card}(\mathrm{set}_x($кортеж$(x, n, A))) = (\mathrm{card}(A))^n)$

Одноэлементное множество, образованное явно заданной функцией

Множество функций, имеющих заданную область определения и заданное выражение для вычисления значений, одноэлементно:

$\forall_{gM}(\mathrm{card}(\mathrm{set}_f(\mathrm{Dom}(f) = M \ \& \ \forall_i(i \in M \to f(i) = g(i)))) = 1)$

Число размещений

Напомним, что размещением называется набор попарно различных элементов множества. Имеются два приема, соответствующих различной идентификации множества размещений:

$\forall_{nA}($конечное$(A) \ \& \ n -$ натуральное $\to \mathrm{card}(\mathrm{set}_x($размещение$(x, n, A))) = \prod_{i=0}^{n-1}(\mathrm{card}(A) - i))$

$\forall_{mnA}($конечное$(A) \ \& \ n -$ целое $\& \ 0 \leq n \ \& \ m = \mathrm{card}(A) \ \& \ 0 \leq m - n \to \mathrm{card}(\mathrm{set}_x($кортеж$(x, n, A) \ \&$ различны$(x))) = m!/(m - n)!)$

Число решений линейного уравнения

$\forall_{abcP}(\neg(a=0)\to \mathrm{card}(\mathrm{set}_x(ax+b=c\ \&\ P(x)))=(1\ \text{при}\ P((c-b)/a),\ \text{иначе}\ 0))$

Число натуральных чисел, делящихся на заданное натуральное число и не превосходящих другого натурального числа

$\forall_{mn}(m-\text{натуральное}\ \&\ n-\text{натуральное}\to \mathrm{card}(\mathrm{set}_i(i-\text{натуральное}\ \&\ n|i\ \&\ i\le m))=[m/n])$

Число множеств, содержащих заданное множество и содержащихся в его надмножестве

$\forall_{ab}(a\subseteq b\ \&\ \text{конечное}(b)\to \mathrm{card}(\mathrm{set}_x(a\subseteq x\ \&\ x\subseteq b\ \&\ x-\mathrm{set}))=2^{\mathrm{card}(b\setminus a)})$

Число множеств заданной мощности, имеющих заданную мощность симметрической разности с другим множеством

$\forall_{ABmn}(A\subseteq B\ \&\ \mathrm{card}(B)=n\ \&\ \mathrm{card}(A)=m\to \mathrm{card}(\mathrm{set}_C(C\subseteq B\ \&\ \mathrm{card}(C)=k\ \&\ \mathrm{card}(A\triangle C)=p))=(C_m^{m+k-p/2}C_{n-m}^{k+p-m/2}\text{при}(m+k-p)-\mathrm{even},\text{иначе}\ 0))$

Уровень срабатывания данного приема равен 1. Чтобы определить число подмножеств C, мощность которых находится в заданном отношении P с мощность симметрической разности $C\triangle A$, добавлен еще один прием, обобщающий предыдущий (его уровень срабатывания равен 2):

$\forall_{ABPmnQ}(A\subseteq B\ \&\ \mathrm{card}(B)=n\ \&\ \mathrm{card}(A)=m\ \&\ (P(i,j)\ \&\ i-\text{целое}\ \&\ j-\text{целое}\ \&\ m+i-j-\mathrm{even}\ \&\ 0\le i\ \&\ 0\le j\ \&\ i\le n\ \&\ j\le n)=(Q(i,j)\ \&\ R(j))\to \mathrm{card}(\mathrm{set}_C(C\subseteq B\ \&\ P(\mathrm{card}(C),\mathrm{card}(b)))=\sum_{i,j,Q(i,j),R(j)}(C_m^{m+i-j/2}C_{n-m}^{i+j-m/2})))$

Указатель "контекст(позиция(x2 теквхожд)вид(x2 симметричразность(x26 x28)))" определяет идентификацию переменной b с выражением $C\triangle A$. Указатель "новаргумент(x40 x28 фикс)" требует, чтобы переменная C входила в отношение $P(\ldots)$ только в виде $\mathrm{card}(C)$ либо $\mathrm{card}(C\triangle A)$. Четвертый антецедент определяет явное разрешение условия $P(i,j)$, сопровожденного рядом простых дополнительных условий на параметры i,j, относительно неизвестной i. Конъюнкция содержащих i членов ответа присваивается $Q(i,j)$, конъюнкция прочих членов - $R(j)$.

Определение числа решений уравнений с помощью пределов и производных

Этот прием подключает к нахождению числа решений уравнения аппарат, созданный в разделе "Математический анализ" для нахождения числа корней функции на заданном множестве. Теорема приема имеет вид:

$\forall_{fgm}(f(x)-\text{число}\ \&\ \mathrm{card}(\mathrm{roots}(\lambda_x(f(x)-g(x),x-\text{число}),\mathbb{R}))=m\to \mathrm{card}(\mathrm{set}_x(x-\text{число}\ \&\ f(x)=g(x)))=m)$

Второй антецедент выделен указателем "идентификатор"; его левая часть обрабатывается вспомогательной задачей на преобразование. Уровень обращения 7 обеспечивает применение в этой задаче указанных выше методов, основанных на пределах и производных.

Мощность множества корней квадратного трехчлена

$\forall_{abc}(\mathrm{card}(\mathrm{set}_x(x - \text{число}\ \&\ ax^2 + bx + c = 0)) = (2\ \text{при}\ \neg(a = 0)\ \&\ 0 < b^2 - 4ac,\ \text{иначе}\ (1\ \text{при}\ a = 0\ \&\ \neg(b = 0) \vee \neg(a = 0)\ \&\ b^2 - 4ac = 0,\ \text{иначе}\ (\infty\ \text{при}\ a = 0\ \&\ b = 0\ \&\ c = 0,\ \text{иначе}\ 0))))$

Случай равенства коэффициента b нулю учитывается указателем "подстановка(фикс(0 1 1 2 2 1 2) x2 0)".

6.3 Разные приемы, используемые при нахождении мощности конечных множеств

Как уже говорилось, определение мощности множества реализовано в решателе как последовательность преобразований выражения card(...). В предыдущем разделе были приведены приемы, используемые на завершающем этапе цепочки преобразований. В данном разделе перечисляются различные "промежуточные" приемы, связанные с рассмотрением конечных множеств.

Усмотрение числового значения мощности

Мощность множества является числом тогда и только тогда, когда это множество конечно. Если в посылках задачи встречается равенство $a = \mathrm{card}(b)$, где a - переменная, причем удается усмотреть конечность множества b, то выводятся посылки, указывающие, что a имеет своим значением целое неотрицательное число:

$\forall_{ab}(a = \mathrm{card}(b)\ \&\ \text{конечное}(b) \rightarrow a - \text{число}\ \&\ a - \text{целое}\ \&\ 0 \leq a)$

Фильтр "не(контекст(посылка(x3)вид(x3 число(x1))))" блокирует применение приема, когда уже имеется посылка число(a). Для исключения утверждений вида "card(a)− число" служит прием, обращающийся к проверочному оператору "усмконечное":

$\forall_a(\text{конечное}(a) \rightarrow \mathrm{card}(a) - \text{число})$

Наконец, для вывода дополнительной посылки о целочисленности мощности конечного множества имеется следующий прием:

$\forall_{ab}(\mathrm{card}(a) = b\ \&\ b - \text{число} \rightarrow b - \text{целое})$

Первый антецедент извлекается из контекста непосредственно; второй - обрабатывается проверочным оператором. Проверяется, что целочисленность значения b до срабатывания приема не усматривается.

Доказательство одноэлементности множества

$\forall_a(a - \mathrm{set} \rightarrow \mathrm{card}(a) = 1 \leftrightarrow \exists_b(b \in a)\ \&\ \forall_{bc}(b \in a\ \&\ c \in a \rightarrow b = c))$

Прием применяется к условию задачи на доказательство. Он выполняет кванторную расшифровку, причем вводит комментарий "кванторнаясвертка", блокирующий обратное преобразование.

Простейшие уравнения с конечной мощностью

Если мощность входит в линейное уравнение, прочие члены которого не содержат символа "мощность" и имеют заведомо числовые значения, то выполняется разрешение относительно данной мощности:

$\forall_{abc}(b - \text{число} \; \& \; c - \text{число} \to \mathrm{card}(a) + b = c \leftrightarrow \mathrm{card}(a) = c - b)$

$\forall_{abc}(b - \text{число} \; \& \; c - \text{число} \to b - \mathrm{card}(a) = c \leftrightarrow \mathrm{card}(a) = b - c)$

Суммирование по множеству значений параметра

Если в описании элементов множества участвует некоторый вспомогательный параметр, то находится множество значений этого параметра, и предпринимается попытка сначала найти мощность подмножества с фиксированным значением параметра, а затем просуммировать такие мощности:

$\forall_{fgA}(\text{функционально}(\mathrm{set}_{xy}(f(x,y) \; \& \; g(x))) \; \& \; \mathrm{set}_y(\exists_x(f(x,y) \; \& \; g(x))) = A \; \& \; \text{конечное}(A) \to \mathrm{card}(\mathrm{set}_x(\exists_y(f(x,y)) \; \& \; g(x))) = \sum_{y, y \in A} \mathrm{card}(\mathrm{set}_x(f(x,y) \; \& \; g(x))))$

Предикат "функционально" означает, что множество наборов не имеет двух различных наборов, у которых совпадают все соответственные (имеющие один и тот же номер) разряды, кроме последнего. Если n - длина наборов, то множество оказывается графиком функции от $n-1$ переменных. Таким образом, первый антецедент проверяет, что параметр y определяется по элементу множества x однозначно, и можно разбить множество на непересекающиеся подмножества с заданными значениями этого параметра. Антецедент обрабатывается вспомогательной задачей на доказательство.

Указатель "кортежпеременных(x23)" разрешает пользоваться приемом для подсчета мощности множеств наборов произвольной длины (в том числе длины 1).

Второй антецедент определяет множество A допустимых значений параметра y. Его левая часть обрабатывается вспомогательной задачей на преобразование. Предварительно конъюнкция под квантором существования разрешается относительно неизвестных x. Так как та же конъюнкция встречается в первом антецеденте и в заменяющей части тождества, где указанное разрешение относительно x нежелательно, введены указатели "копия". Они определяют использование в этих местах ненормализованной конъюнкции.

Для частного случая, когда параметром набора служит его поднабор, полученный исключением элемента с заданным номером, введен дополнительный прием:

$\forall_{fghmA}(f(x,y) = (g(x,y) \; \& \; h(y)) \; \& \; m(y) = \mathrm{card}(\mathrm{set}_x(g(x,y))) \to \mathrm{card}(\mathrm{set}_{xy}(f(x,y))) = \sum_{y, h(y)} m(y))$

Указателем "кортежпеременных" здесь выделена переменная y - она и играет роль параметра. Первый антецедент разрешает условие $f(x,y)$ принадлежности набора множеству относительно переменной x. При этом $h(y)$ - конъюнкция всех не зависящих от x членов ответа; $g(x,y)$ - остальные члены ответа. Второй антецедент выделен указателем "идентификатор". Правая часть его обрабатывается вспомогательной задачей на преобразование, которой придается дополнительная посылка $h(y)$. Таким образом, выражению $m(y)$ присваивается мощность подмножества, соответствующего заданному значению y. Фильтры проверяют, что результирующее выражение для $m(y)$ не содержит символа "мощность".

Еще два приема суммирования относятся к случаю подсчета числа пар, причем параметром является один из элементов пары:

$\forall_{mnPpq}(\mathrm{set}_j(0 \le n(j) - m(j)\ \&\ P(j)) = \{p,\dots,q\} \to \mathrm{card}(\mathrm{set}_{ij}(i \in \{m(j),\dots,n(j)\}\ \&\ P(j))) = \sum_{j=p}^{q}(n(j) - m(j) + 1))$

$\forall_{nPpq}(\mathrm{set}_j(0 \le n(j) - 1\ \&\ P(j)) = \{p,\dots,q\} \to \mathrm{card}(\mathrm{set}_{ij}(i - \text{натуральное}\ \&\ i \le n(j)\ \&\ P(j))) = \sum_{j=p}^{q} n(j))$

В обоих случаях переменная j выделена указателем "кортежпеременных", хотя видно, что ее значениями служат целые числа, и, следовательно, идентифицироваться она должна с единственной переменной. Более того, фильтр "длинатекста(x10 1)" явно указывает, что эта переменная должна быть единственной. Однако, выбранный способ характеризации переменной j позволяет идентифицировать переменную i с произвольным - первым либо вторым - элементом пары. При отсутствии указателя "кортежпеременных" i обязательно идентифицировалась бы с первой переменной. В остальном приемы достаточно стандартны.

Разбиение семейства множеств на подсемейства множеств одинаковой мощности

Фактически приемы этого раздела относятся к частному случаю суммирования по множеству значений параметра. Здесь перечисляются наборы, некоторый элемент которых представляет собой множество, а роль параметра суммирования играет мощность данного множества.

$\forall_{APQM}(M = \mathrm{set}_i(P(i)\ \&\ i\text{−целое}\ \&\ 0 \le i\ \&\ i \le \mathrm{card}(A))\ \&\ \text{конечное}(A) \to \mathrm{card}(\mathrm{set}_{xy}(x\text{−}\mathrm{set}\ \&\ x \subseteq A\ \&\ P(\mathrm{card}(x))\ \&\ Q(x,y))) = \sum_{i,i \in M} \mathrm{card}(\mathrm{set}_{xy}(x - \mathrm{set}\ \&\ x \subseteq A\ \&\ \mathrm{card}(x) = i\ \&\ Q(x,y))))$

Переменная y выделена указателем "кортежпеременных" и может идентифицироваться с набором переменных произвольной длины, в том числе с пустым. Указатель "новаргумент(x40 x23 фикс)" означает, что условие $P(\mathrm{card}(x))$ содержит переменную x только в подвыражениях $\mathrm{card}(x)$. При этом указатель "перечень(x40 и(вхождениетерма(x40 мощность(x23))не(пересекаются(x24 x40))))" определяет комплектование P из всех конъюнктивных членов описателя, не имеющих вхождений переменных y, но имеющих вхождение выражения $\mathrm{card}(x)$. Первый антецедент выделен указателем "идентификатор", причем его правая часть обрабатывается вспомогательной задачей на преобразование. Он служит для определения множества M допустимых значений мощности множества x. Проверяется, что заголовком выражения M служит символ "номера" либо "перечень". Кроме того, отсекаются вырожденные случаи, когда P имеет вид равенства, фиксирующего мощность множества x, либо является константой "истина".

Следующий прием аналогичен только что рассмотренному, но имеет несколько другой вид описателя "класс", где условие на мощность x сразу дает возможность определить диапазон суммирования:

$\forall_{Qmnf}(0 \le f(y) \to \mathrm{card}(\mathrm{set}_{xy}(x\text{−}\mathrm{set}\ \&\ m \le \mathrm{card}(x)\ \&\ \mathrm{card}(x) + f(y) - n = 0\ \&\ Q(x,y))) = \sum_{i=m}^{n} \mathrm{card}(\mathrm{set}_{xy}(x - \mathrm{set}\ \&\ \mathrm{card}(x) = i\ \&\ i + f(y) - n = 0\ \&\ Q(x,y))))$

Еще один специальный случай - мощность множества x явно выражена через параметры y, причем в остальных утверждениях под описателем x и y встречаются лишь раздельно. Это позволяет при суммировании провести разделение переменных:

$\forall_{ABtfg}((z = t(y)\ \&\ z - \text{целое}\ \&\ B(y)) = (f(y,z)\ \&\ g(z)) \rightarrow \text{card}(\text{set}_{xy}(\text{card}(x) = t(y)\ \&\ A(x)\ \&\ B(y))) = \sum_{z,g(z)}(\text{card}(\text{set}_y(f(y,z)))\text{card}(\text{set}_x(\text{card}(x) = z\ \&\ A(x)))))$

Первый антецедент разрешает относительно y условие равенства мощности $t(y)$ множества x, определяемой по значениям y, заданной величине z. Здесь z - новая переменная. Те члены ответа, которые не зависят от y, объединяются в задающее область суммирования условие $g(z)$; прочие - в условие $f(y,z)$. Фильтр "не(заголовок(х24 пустоеслово))" отсекает вырожденный случай пустого множества переменных y.

Наконец, имеется прием для случая, когда множество x пусто либо одноэлементно:

$\forall_{AP}(\text{card}(\text{set}_{xy}(x - \text{set}\ \&\ x \subseteq A\ \&\ \text{card} \leq 1\ \&\ P(x,y))) = \text{card}(\text{set}_y(P(\emptyset,y))) = \text{card}(\text{set}_{xy}(x\ inA\ \&\ P(\{x\},y))))$

Переход к суммированию мощностей дополнительного семейства подмножеств

Прием сводит подсчет числа подмножеств, мощность которых не менее заданной величины, к подсчету числа подмножеств, мощность которых не более заданной величины:

$\forall_{An}(\text{конечное}(A)\ \&\ 0 < \text{card}(A) - 2n \rightarrow \text{card}(\text{set}_x(x \subseteq A\ \&\ x - \text{set}\ \&\ n \leq \text{card}(x))) = 2^{\text{card}(A)} - \text{card}(\text{set}_x(x \subseteq A\ \&\ x - \text{set}\ \&\ \text{card}(x) \leq n - 1)))$

Второй антецедент введен для того, чтобы в случае значений n, больших половины мощности A, сразу переходить к подсчету числа их дополнений до A.

Мощность прямого произведения

Для случая двух сомножителей имеются два приема. Первый из них предпринимает попытку вычислить мощности сомножителей с помощью вспомогательных задач; второй (его уровень срабатывания больше) - просто заменяет исходное выражение на произведение пока еще не вычисленных мощностей сомножителей.

$\forall_{abmn}(m = \text{card}(a)\ \&\ n = \text{card}(b)\ \&\ m - \text{число}\ \&\ n - \text{число} \rightarrow \text{card}(a \times b) = mn)$

$\forall_{AB}(\text{конечное}(A)\ \&\ \text{конечное}(B) \rightarrow \text{card}(A \times B) = \text{card}(A)\text{card}(B))$

Для случая прямого произведения конечного набора множеств создан еще один прием:

$\forall_{An}(\text{card}(\prod_{i=1}^{n} A(i) = \prod_{i=1}^{n} \text{card}A(i)))$

Здесь левый знак конечного произведения соответствует символу "Прямоепроизведение" для умножения множеств; правый - символу "прямоепроизведение" для обычного конечного произведения. Указатель "развертка" отсутствует.

Учет условия, имеющего своим заголовком отрицание

Если среди утверждений, описывающих условие принадлежности элемента множеству, встречается отрицание некоторого утверждения P, то множество представляется в виде разности:

$\forall_{fgab}(a = \text{card}(\text{set}_x(g(x)))\ \&\ b = \text{card}(\text{set}_x(f(x)\ \&\ g(x)))\ \&\ a - \text{число}\ \&\ b - \text{число} \rightarrow \text{card}(\text{set}_x(\neg(f(x))\ \&\ g(x))) = a - b)$

Мощности a, b вычисляются путем обращения к вспомогательным задачам на преобразование.

Попытка рассмотрения дополнения в случае условия существования

Если в описании множества встречается утверждение существования, то может оказаться полезным прием, аналогичный предыдущему: множество представляется в виде разности, где вычитаемое соответствует отрицанию квантора существования. Имеются два приема такого типа. Первый, для общего случая, срабатывает на уровне 5. Второй связан с частным случаем - множеством отображений, где уровень срабатывания оказалось целесообразным понизить до 2.

$\forall_{abPQB}(\mathrm{card}(\mathrm{set}_M(Q(M))) = a \;\&\; \mathrm{card}(\mathrm{set}_M(\forall_i(i \in B \to \neg(P(i,M))) \;\&\; Q(M))) = b \;\&\; a - \text{число} \;\&\; b - \text{число} \to \mathrm{card}(\mathrm{set}_M(\exists_i(i \in B \;\&\; P(i,M)) \;\&\; Q(M))) = a - b)$

$\forall_{abPQB}(\mathrm{card}(\mathrm{set}_M(\text{Отображение}(M,B,C) \;\&\; Q(M))) = a \;\&\; \mathrm{card}(\mathrm{set}_M(\forall_i(i \in B \to \neg(P(i,M))) \;\&\; \text{Отображение}(M,B,C) \;\&\; Q(M))) = b \;\&\; a - \text{число} \;\&\; b - \text{число} \to \mathrm{card}(\mathrm{set}_M(\exists_i(i \in B \;\&\; P(i,M)) \;\&\; \text{Отображение}(M,B,C) \;\&\; Q(M))) = a - b)$

В обоих случаях мощности a, b вычисляются с помощью вспомогательных задач на преобразование, причем фильтры приемов проверяют, что выражения a, b не содержат символа "мощность".

Мощность объединения

Если удается вычислить мощность a пересечения двух множеств, то мощность объединения заменяется на сумму мощностей этих множеств, уменьшенную на a:

$\forall_{ABa}(\text{конечное}(A) \;\&\; \text{конечное}(B) \;\&\; a = \mathrm{card}(A \cap B) \to \mathrm{card}(A \cup B) = \mathrm{card}(A) + \mathrm{card}(B) - a)$

Проверка того, что мощность пересечения найдена, выполняется фильтром "не(входит(мощность x1))".

Представление мощности в виде произведения при независимости числа элементов от значения параметра

Если элемент рассматриваемого множества задается при участии некоторого параметра, причем число таких элементов для различных значений параметра неизменно, то мощность множества представляется как произведение числа допустимых значений параметра на мощность подмножества, получаемого при фиксации значения параметра. Прием для общего случая срабатывает на уровне 9:

$\forall_{PQa}(a = \mathrm{card}(\mathrm{set}_j(Q(i,j))) \to \mathrm{card}(\mathrm{set}_{ij}(P(i) \;\&\; Q(i,j))) = \mathrm{card}(\mathrm{set}_i(P(i)))a)$

Роль параметра играет переменная i; переменная j выделена указателем "кортежпеременных" и может идентифицироваться с набором переменных произвольной ненулевой длины. Фильтр "не(входит(x9 x1))" проверяет независимость мощности a от i. Эта мощность вычисляется путем обращения к вспомогательной задаче на преобразование. Прием имеет указатель "лимит(1000000)", устанавливающий "средний" уровень трудоемкости для попытки применения приема.

В частном случае, когда роль параметра играют подмножества заданной мощности некоторого заданного множества, используется дополнительный прием, срабатывающий на уровне 3:

$\forall_{Afp}(p = \mathrm{card}(\mathrm{set}_y(f(x,y))) \to \mathrm{card}(\mathrm{set}_{xy}(x \subseteq A \;\&\; x - \mathrm{set} \;\&\; \mathrm{card}(x) = m \;\&\; f(x,y))) = C^m_{\mathrm{card}(A)}p)$

Как и выше, независимость мощности p от значений параметра устанавливается фильтром вида "не(входит(x23 x15))".

Число выборок, в которых наложены ограничения на подмножества элементов заданных типов

Пусть на множестве A задано отображение f, сопоставляющее каждому элементу его тип - некоторое натуральное число. Если требуется подсчитать число таких подмножеств множества A, у которых элементы каждого типа образуют класс, удовлетворяющий условию P, то достаточно перемножить количества подмножеств элементов каждого типа, удовлетворяющих P:

$\forall_{AfmP}(\text{Отображение}(f, A, \{1, \ldots, m\}) \to \text{card}(\text{set}_B(B \subseteq A\ \&\ B - \text{set}\ \&\ \forall_i(i \in \{1, \ldots, m\} \to P(\text{слой}(\text{сужение}(f, B), i)))))) = \prod_{i=1}^{m} \text{card}(\text{set}_y(y \subseteq \text{слой}(f, i)\ \&\ y - \text{set}\ \& P(y))))$

Указатель "новаргумент(x40 x27 фикс)" определяет способ идентификации утверждения $P(\ldots)$.

Суммирование мощностей классов разбиения

Снова рассматриваем на множестве A отображение, определяющие типы элементов. Если нужно вычислить мощность подмножества B множества A, для которого известны количества элементов каждого типа, то эти количества суммируются:

$\forall_{fABmq}(B \subseteq A\ \&\ \text{Отображение}(f, A, \{1, \ldots, m\})\ \&\ \forall_i(i \in \{1, \ldots, m\} \to \text{card}(\text{слой}(\text{сужение}(f, B), i)) = q(i)) \to \text{card}(B) = \sum_{i=1}^{m} q(i))$

Первый антецедент выделен указателем "блокпроверок", остальные два - присутствуют в контексте замены явным образом. Проверяется, что выражение $q(i)$ не содержит символа "мощность".

Учет перестановок на множестве значений отображения, если прообразы элементов имеют попарно различные характеристики

Если требуется найти число отображений конечного множества A в множество $\{1, \ldots, n\}$, у которых с точностью до перестановок заданы попарно различные характеристики прообразов элементов, то определяется число этих отображений при какой-либо фиксации перестановки, и домножается на число перестановок:

$\forall_{AnPQh}(\text{взаимнооднозначно}(h)\ \&\ \text{конечное}(A)\ \&\ n - \text{натуральное} \to \text{card}(\text{set}_f(\text{Отображение}(f, A, \{1, \ldots, n\})\ \&\ P(f)\ \&\ \exists_g(\text{перестановка}(g, \{1, \ldots, n\})\ \&\ g - \text{функция}\ \&\ \forall_i(i \in \{1, \ldots, n\} \to Q(\text{слой}(f, g(i))) = h(i)))))) = n!\text{card}(\text{set}_f(\text{Отображение}(f, A, B)\ \&\ P(f)\ \&\ \forall_i(i \in \{1, \ldots, n\} \to Q(\text{слой}(f, i)) = h(i)))))$

Известные мощности классов разбиения области значений

Пусть имеется отношение $Q(f, x, y)$, определяющее однозначно некоторую характеристику x значения y функции f. Пусть, далее, нужно найти число отображений f конечного множества A в конечное множество B, у которых известны мощности $k(x)$ подмножеств области значений с заданным значением характеристики x. Если удается усмотреть, что число d таких отображений при каком-то фиксированном задании характеристик $m(y)$, удовлетворяющих перечисленным условиям, не зависит от конкретного выбора функции m, то для получения результата используются соображения симметрии: величина d умножается на число перестановок множества B и делится на произведение чисел перестановок в каждом классе элементов, имеющих фиксированную характеристику. Теорема приема имеет следующий вид:

$\forall_{ABPQkmd}(\text{функционально}(\text{set}_{yx}(P(x)\ \&\ y \in B\ \&\ Q(f,x,y)))\ \&\ \text{конечное}(B)\ \&\ \text{конечное}(A)\ \&\ \forall_y(y \in B \to \exists_x(P(x)\ \&\ Q(f,x,y)))\ \&\ \text{card}(\text{set}_f(\text{Отображение}(f,A,B)\ \&\ \forall_y(y \in B \to Q(f,m(y),y)))) = d\ \&\ d - \text{число} \to \text{card}(\text{set}_f(\text{Отображение}(f,A,B)\ \&\ \forall_x(P(x) \to \text{card}(\text{set}_y(y \in B\ \&\ Q(f,x,y))) = k(x)))) = (d \cdot \text{card}(B)!/\prod_{x,P(x)} k(x)!))$

Первый антецедент проверяет однозначность определения характеристики x. Здесь $P(x)$ - условие на область допустимых значений характеристики. Условие независимости определяемой в пятом антецеденте мощности d от m обеспечивается фильтром "не(входит(x13 x4))". Вспомогательная задача, вычисляющая d, получает необходимые дополнительные посылки, ограничивающие требуемым образом функцию m.

Сведение к множеству двоичных наборов меньшей размерности

Если требуется найти число двоичных наборов длины n, у которых фиксировано значение a i-го разряда, причем прочие условия на набор зависят только от числа его разрядов, равных b, то предпринимается переход к наборам длины $n-1$:

$\forall_{abnP}(\text{card}(\text{set}_x(\text{двнабор}(x,n)\ \&\ x(i)=a\ \&\ P(\text{колич}(x,b)))) = \text{card}(\text{set}_x(\text{двнабор}(x, n-1)\ \&\ P(\text{колич}(x,b) + (1 \text{ при } a=b, \text{иначе } 0)))))$

Напомним, что колич(x,b) - число разрядов набора x, равных b. Указатель "контекст(подчинено(x3 тевхожд)вид(x3 колич(x23 x2)))" обеспечивает первоначальное усмотрение выражения колич(x,b) в преобразуемом выражении. Указатель "новаргумент(x40 x23 фикс)" определяет идентификацию P(колич(x,b)) и проверку того, что x входит в идентифицирующий терм только в виде колич(x,b).

Множество пар с фиксированным элементом

Если один элемент пары фиксирован, а прочие условия на пару относятся к другому элементу, то пара исключается:

$\forall_{ijaP}(\neg(i-j=0) \to \text{card}(\text{set}_x(l(x)=2\ \&\ x - \text{слово}\ \&\ x(i)=a\ \&\ P(x(j)))) = \text{card}(\text{set}_x(P(x))))$

Как и в предыдущем приеме, используется указатель дополнительной идентификации "контекст(позиция(x2 теквхожд)вид(x2 значение(x23 x10))не(равно(x9 x10)))", обеспечивающий усмотрение выражения $x(j)$.

Усмотрение равенства вложенных множеств из сравнения мощностей

$\forall_{AB}(\text{конечное}(B)\ \&\ \text{card}(A) - \text{card}(B) = 0 \to A \subseteq B \leftrightarrow A = B)$

Прием заменяет включение множеств на их равенство, если это включение расположено внутри описателя "класс".

Число множеств, включающихся в одно множество и не включающихся в другое

Прием переходит к разности двух количеств подмножеств:

$\forall_{ABP}(\mathrm{card}(\mathrm{set}_x(x - \mathrm{set}\ \&\ x \subseteq A\ \&\ P(x)\ \&\ \neg(x \subseteq B))) = \mathrm{card}(\mathrm{set}_x(x - \mathrm{set}\ \&\ x \subseteq A\ \&\ P(x))) - \mathrm{card}(\mathrm{set}_x(x - \mathrm{set}\ \&\ x \subseteq A \cap B\ \&\ P(x))))$

Переход от графиков отображений к отображениям

Если определяется мощность множества m - элементных подмножеств прямого произведения, являющихся графиками отображений, то выполняется переход к множеству отображений, определенных на m - элементных подмножествах первого сомножителя:

$\forall_{ABm}(\mathrm{card}(\mathrm{set}_x(x - \mathrm{set}\ \&\ x \subseteq A \times B\ \&\ \mathrm{card}(x) = m\ \&\ \forall_{ij}(i(2) = j(2)\ \&\ \neg(i = j)\ \&\ i \in x \to \neg(j \in x)))) = \mathrm{card}(\mathrm{set}_{fC}(C \subseteq B\ \&\ \mathrm{card}(C) = m\ \&\ \text{Отображение}(f, C, A))))$

Указатель "смравно(фикс(0 1 1 2 2 2))" предусматривает возможность идентификации прямого произведения $A \times B$ либо с произвольным выражением T, для которого есть посылка вида $T = A' \times B'$, либо непосредственно с прямым произведением.

Аналогичный прием введен для "перевернутых" графиков.

Усмотрение условия на симметрическую разность множеств

Если находится число наборов, в которых упроминаются два подмножества x, y заданного множества A, причем прочие условия на x, y выражаются только через их симметрическую разность, то появляется возможность декомпозиции, основанная на представлении x как симметрической разности y и $x \bigtriangleup y$:

$\forall_{APQ}(\mathrm{card}(\mathrm{set}_{xyz}(x - \mathrm{set}\ \&\ x \subseteq A\ \&\ y - \mathrm{set}\ \&\ y \subseteq A\ \&\ P(x \bigtriangleup y)\ \&\ Q(y, z))) = \mathrm{card}(\mathrm{set}_x(x - \mathrm{set}\ \&\ x \subseteq A\ \&\ P(x)))\mathrm{card}(\mathrm{set}_{yz}(y - \mathrm{set}\ \&\ y \subseteq A\ \&\ Q(y, z))))$

Чтобы идентифицировать утверждение $P(x \bigtriangleup y)$, введен указатель "новаргумент(x40 набор(x23 x24)извлечмножества)". Нормализатор "извлечмножества" создан, чтобы усматривать мощность симметрической разности в неявных случаях. Пока он имеет единственный прием:

$\forall_{abc}(\mathrm{card}(a \setminus b) + \mathrm{card}(b \setminus a) + c = \mathrm{card}(a \bigtriangleup b) + c)$.

Еще один прием, связанный с использованием тождества $x = (x \bigtriangleup y) \bigtriangleup y)$, позволяет получить под описателем утверждение, явно задающее мощность одного из рассматриваемых подмножеств:

$\forall_{APQ}(\mathrm{card}(\mathrm{set}_{xyz}(x - \mathrm{set}\ \&\ x \subseteq A\ \&\ y - \mathrm{set}\ \&\ y \subseteq A\ \&\ P(x \bigtriangleup y)\ \&\ Q(x, y, z))) = \mathrm{card}(\mathrm{set}_{xyz}(x - \mathrm{set}\ \&\ x \subseteq A\ \&\ y - \mathrm{set}\ \&\ y \subseteq A\ \&\ P(x)\ \&\ Q(x \bigtriangleup y, y, z))))$

Здесь $P(\ldots)$ идентифицируется с единственным содержащим одновременно x, y утверждением, причем это утверждение оказывается равенством вида $\mathrm{card}(x \bigtriangleup y) + m = n$, где m, n не содержат x, y.

Раздельное рассмотрение внешних и внутренних элементов

Если подсчитывается число элементов множества A, удовлетворяющих некоторому условию $P(x)$, причем внутри данного условия встречается разность $y \setminus \{x\}$, то предпринимается попытка отдельно найти число указанных элементов внутри подмножества y и вне этого подмножества:

$\forall_{APy}(y \subseteq A\ \&\ m = \mathrm{card}(\mathrm{set}_x(x \in A\ \&\ P(x)\ \&\ x \in y))\ \&\ n = \mathrm{card}(\mathrm{set}_x(x \in A\ \&\ P(x)\ \&\ \neg(x \in y))) \to \mathrm{card}(\mathrm{set}_x(x \in A\ \&\ P(x))) = m + n)$

Указатель "контекст(позиция(x1 фикс(0 1 1 2 2))вид(x1 разность(x24 перечень(набор(x23)))))" определяет идентификацию подмножества y. Фильтры "не(входит(мощность x13))", "не(входит(мощность x14))" проверяют, что предпринимаемые с помощью задач на преобразование попытки вычислить мощность вспомогательных подмножеств оказались удачными.

Подмножество делителей числа, разложенного на простые множители

Если множество чисел выражено через наборы степеней в их разложении на заданные простые множители, то выполняется переход к мощности множества этих наборов:

$\forall_{parQ}(\text{простое}(p(i)) \to \mathrm{card}(\mathrm{set}_m(\exists_b(m = \prod_{i=1}^r (p(i))^{b(i)} \ \& \ Q(b)))) = \mathrm{card}(\mathrm{set}_b(Q(b))))$

Первый антецедент обрабатывается проверочным оператором, причем указатель "занесениепосылки(1 принадлежит(x9 номера(1 x17)))" добавляет к контексту проверки утверждение $i \in \{1, \ldots, r\}$. Переменные p, Q выделены указателем "отображение".

Мощность множества неупорядоченных пар

Если условие на перечисляемые пары не изменяется при перестановке их элементов, причем элементы эти различны, то вычисляется число упорядоченных пар, которое затем делится на 2:

$$\forall_Q(Q(z,y) \ \& \ \neg(y = z) \to \mathrm{card}(\mathrm{set}_x(\exists_{yz}(x = \{y,z\} \ \& \ Q(y,z)))) = \frac{\mathrm{card}(\mathrm{set}_{yz}(Q(y,z)))}{2})$$

Антецеденты обрабатываются вспомогательными задачами на доказательство, которым придается дополнительная посылка $Q(y, z)$).

Коррекция условия на целочисленный параметр

Следующий прием выполняет стандартизацию, обеспечивающую представление множества пар индексов:

$\forall_{mnk}(\mathrm{set}_{ij}(i \in \{m, \ldots, j+k\} \ \& \ j - \text{целое} \ \& \ j \leq n) = \mathrm{set}_{ij}(i \in \{m, \ldots, j+k\} \ \& \ j \in \{k-m, \ldots, n\}))$

6.4 Бесконечные множества

Приемы для работы с мощностями бесконечных множеств возникали при обучении решателя лишь эпизодически, в самых простых случаях. Здесь собраны несколько таких приемов:

Объединение счетного и конечного множеств

$\forall_{ab}(\mathrm{card}(a) = \text{счетное} \ \& \ \text{конечное}(b) \to \mathrm{card}(a \cup b) = \text{счетное})$

Определение мощности множества a выполняется нормализатором "нормощность".

Разность бесконечного и конечного множеств

$\forall_{ab}(\neg(\text{конечное}(a))\ \&\ \text{конечное}(b) \to \text{card}(a \setminus b) = \text{card}(a))$

Оба антецедента обрабатываются проверочными операторами.

Исключение равенства мощности бесконечности

Если при первоначальной постановке задачи возникает утверждение вида $\text{card}(a) = \infty$, то оно немедленно преобразуется к виду $\neg(\text{конечное}(a))$. Фактически, здесь прием относится к интерфейсу решателя.

6.5 Переход к равномощному множеству

Одна из наиболее часто используемых в комбинаторике разновидностей приемов - переходы к равномощным множествам. Обычно они упрощают условие под описателем либо приводят его к стандартному виду, облегчающему вычисление мощности.

Устранение параметра, явно выраженного через другие параметры

Если подсчитывается количество наборов значений параметров, удовлетворяющих некоторым условиям, причем один из параметров однозначно определяется через остальные (быть может, с привлечением вспомогательных параметров), то он отбрасывается (заменяется на вспомогательные параметры), с соответствующей коррекцией условий на перечисляемые наборы. На этом принципе основаны нижеследующие приемы:

1. $\forall_{fg}(\text{функционально}(\text{set}_{xz}(\exists_y(z = y\ \&\ x = f(w,y)\ \&\ g(w,y)))) \to \text{card}(\text{set}_{wx}(\exists_y(x = f(w,y)\ \&\ g(w,y)))) = \text{card}(\text{set}_{wy}(g(w,y))))$

   Параметр x выражается через оставшиеся параметры w перечисляемых наборов, а также через вспомогательный параметр y. Первый антецедент проверяет, что y восстанавливается по x, w однозначно. После этого перечисляются наборы, у которых место "основного" параметра x занял вспомогательный параметр y. Условие, накладываемое на новые наборы, проще исходного.

2. $\forall_{fAgh}(A \cap \text{Dom}(f(z)) = \emptyset \to \text{card}(\text{set}_{wx}(\exists_{yz}(x = \text{доопределение}(f(z), A, y)\ \&\ g(w,y,z))\ \&\ h(w))) = \text{card}(\text{set}_{wxy}(\exists_z(x = f(z)\ \&\ g(w,y,z))\ \&\ h(w))))$

   Параметр x оказывается результатом доопределения функции $f(z)$ на множестве A, не пересекающемся с ее областью определения, значением y. Параметры y, z - вспомогательные; они связаны с основными параметрами w условием $g(w,y,z)$. Прием заменяет параметр x парой новых параметров, первый из которых (он по-прежнему обозначаен через x) - функция $f(z)$, второй - вспомогательный параметр y. Такая замена приводит к упрощению условия на перечисляемые наборы.

3. $\forall_{abpqrf}(\neg(a = 0)\ \&\ \neg(q = 0)\ \&\ p(n) = an+b \to \text{card}(\text{set}_x(\exists_n(x = p(n)q/r\ \&\ f(n)))) = \text{card}(\text{set}_n(f(n))))$

   При переходе от перечисления значений x к перечислению значений n используется взаимная однозначность невырожденного линейного отображения.

4. $\forall_{fg}(\mathrm{card}(\mathrm{set}_{xy}(x = f(y)\ \&\ g(y))) = \mathrm{card}(\mathrm{set}_y(g(y))))$
5. $\forall_{aP}(\mathrm{card}(\mathrm{set}_x(\exists_{yz}(x = \text{наложение}(y, z, a)\ \&\ P(y, z)))) = \mathrm{card}(\mathrm{set}_{yz}(P(y, z))))$

   Набор x однозначным определяет составляющие его наборы y, z: номера позиций, на которых находятся элементы y, образуют фиксированный список a. Поэтому задача сводится к подсчету пар наборов y, z.
6. $\forall_{abP}(\mathrm{card}(\mathrm{set}_x(\exists_y(x = \text{наложение}(a, y, b)\ \&\ P(y)))) = \mathrm{card}(\mathrm{set}_y(P(y))))$

Переход к области определения - начальному отрезку натуральных чисел

Если подсчитывается число отображений, определенных на конечных множествах, причем дополнительные условия на отображение сводятся лишь к мощностям его прообразов, то предпринимается переход к рассмотрению отображений, определенных на начальном отрезке натурального ряда:

$\forall_{bcdpABg}(\mathrm{card}(A(b, d)) = p(d)\ \&\ p(d) - \text{число} \to \mathrm{card}(\mathrm{set}_{bcd}(\text{Отображение}(c, A(b, d), B(d))\ \&\ g(b, c, d))) = \mathrm{card}(\mathrm{set}_{bcd}(\text{Отображение}(c, \{1, \ldots, p(d)\}, B(d))\ \&\ g(b, c, d))))$

Фильтр "не(контекст(разряд(значение(x7 набор(x2 x3 x4))x3 x5)и(не(контекст(подтерм(мощность(слой(теквхожд(x5)x9))))) не(контекст(подтерм(мощность(прообраз(теквхожд(x5)x9))))))))" разрешает лишь такие вхождения c в дополнительное условие $g(b, c, d)$, которые имеют вид card(слой(c, y)), card(прообраз(c, y)). Первый антецедент при вычислении мощности $p(d)$ области определения использует вспомогательную задачу.

Переход от условия включения одноэлементного подмножества к условию принадлежности элемента

$\forall_{Bf}(\mathrm{card}(\mathrm{set}_{ax}(\mathrm{card}(a) = 1\ \&\ a \subseteq B\ \&\ f(a, x))) = \mathrm{card}(\mathrm{set}_{ax}(a \in B\ \&\ f(\{a\}, x))))$

Нормировка целочисленного параметра

$\forall_{mnfg}(m - \text{целое}\ \&\ n - \text{целое} \to \mathrm{card}(\mathrm{set}_{xy}(x - \text{целое}\ \&\ 0 \leq x + m\ \&\ x + f(y) = n\ \&\ g(y))) = \mathrm{card}(\mathrm{set}_{xy}(x - \text{целое}\ \&\ 0 \leq x\ \&\ x + f(y) = n + m\ \&\ g(y))))$

Прием выполняет переход от целочисленного параметра с заданной нижней границей к неотрицательному параметру.

Отбрасывание коэффициента элемента пары

$\forall_{afgA}(\neg(a = 0) \to \mathrm{card}(\mathrm{set}_x(\exists_y(x = (af(y), g(y))\ \&\ A(y)))) = \mathrm{card}(\mathrm{set}_x(\exists_y(x = (f(y), g(y))\ \&\ A(y)))))$

Чтобы прием отбрасывал коэффициент как у первого, так и второго элементов пар, введен указатель "модифсборка(фикс(0 1 1 2 2 1 2)фикс(0 2 1 2 2 1 2))". Он разрешает перестановку операндов набора по первому указателю вхождения "фикс(...)", выполняя такую же перестановку для операндов набора по второму указателю.

Перенесение области определения функции из вспомогательных параметров в основные

Так как область определения отображения восстанавливается однозначно, ее можно перенести из вспомогательных параметров в основные, устранив при этом квантор существования:

$\forall_{ghB}(\mathrm{card}(\mathrm{set}_{fx}(\exists_A(g(A,f,x)\ \&\ \text{Отображение}(f,A,B(x)))\ \&\ h(f,x))) = \mathrm{card}(\mathrm{set}_{fxA}(g(A,f,x)\ \&\ \text{Отображение}(f,A,B(x))\ \&\ h(f,x))))$

Использование параметрических описаний

В некоторых случаях бывает полезно выразить один из параметров перечисляемых наборов через вспомогательные параметры, однозначно по нему определяемые:

1. Включение известного множества. Надмножество представляется в виде объединения этого множества и некоторого не пересекающегося с ним вспомогательного множества:

   $\forall_{fg}(\mathrm{card}(\mathrm{set}_{xy}(x - \mathrm{set}\ \&\ f(x,y)\ \&\ g \subseteq x)) = \mathrm{card}(\mathrm{set}_{xy}(x - \mathrm{set}\ \&\ g \cap x = \emptyset\ \&\ f(x \cup g, y))))$

2. Известная мощность пересечения с известным множеством. Пусть множество x содержится в некотором заданном множестве $A(v)$, причем мощность пересечения его с другим множеством $B(v)$ равна m. Тогда представляем x в виде $y \cup z$, где z является m - элементным подмножеством множества $A(v) \cap B(v)$, а y - произвольным подмножеством множества $A(v) \setminus B(v)$:

   $\forall_{ABmf}(\mathrm{card}(\mathrm{set}_{xv}(x \subseteq A(v)\ \&\ \mathrm{card}(x \cap B(v)) = m\ \&\ f(x,v))) = \mathrm{card}(\mathrm{set}_{yzv}(y - \mathrm{set}\ \&\ z - \mathrm{set}\ \&\ y \subseteq A(v) \setminus B(v)\ \&\ z \subseteq A(v) \cap B(v)\ \&\ \mathrm{card}(z) = m\ \&\ f(y \cup z, v))))$

3. Функция с известной мощностью прообраза элемента. Пусть f - отображение некоторого множества $A(x)$ в множество $B(x)$, причем мощность прообраза элемента $b(x)$ равна n. Тогда выбирается произвольное n - элементное подмножество c множества $A(x)$, и f представляется как доопределение некоторого вспомогательного отображения из $A(x) \setminus c$ в $B(x) \setminus \{b(x)\}$ (оно ниже обозначено той же переменной f) значением $b(x)$ во всех точках множества c:

   $\forall_{ABgbn}(\mathrm{card}(\mathrm{set}_{fx}(\text{Отображение}(f,A(x),B(x))\ \&\ \mathrm{card}(\text{слой}(f,b(x))) = n\ \&\ g(f,x))) = \mathrm{card}(\mathrm{set}_{fxc}(c - \mathrm{set}\ \&\ c \subseteq A(x)\ \&\ \mathrm{card}(c) = n\ \&\ \text{Отображение}(f, A(x) \setminus c, B(x) \setminus \{b(x)\})\ \&\ g(\text{доопределение}(f,c,b(x)),x))))$

4. Функция с известной мощностью прообраза множества. Если в предыдущей ситуации вместо прообраза элемента рассматривается прообраз множества, то возникает аналогичный прием:

   $\forall_{ABgbn}(\mathrm{card}(\mathrm{set}_{fx}(\text{Отображение}(f,A(x),B(x))\ \&\ \mathrm{card}(\text{прообраз}(f,b(x))) = n\ \&\ g(f,x))) = \mathrm{card}(\mathrm{set}_{fxch}(c - \mathrm{set}\ \&\ c \subseteq A(x)\ \&\ \mathrm{card}(c) = n\ \&\ \text{Отображение}(f, A(x) \setminus c, B(x) \setminus b(x))\ \&\ \text{Отображение}(h,c,b(x))\ \&\ g(\text{таблица}\{f,h\},x))))$

5. Функция с условием на значение в конкретной точке.

   Если часть условия на перечисляемые взаимно-однозначные отображения f с областью определения $A(x)$ и множеством значений $B(x)$ сводится к ограничению на значение b отображения f в точке a, то в $B(x)$ выбирается произвольный удовлетворяющий данному ограничению элемент b, и отображение f

представляется как результат доопределения некоторого взаимно-однозначного отображения g множества $A(x) \setminus \{a\}$ на $B(x) \setminus \{b\}$ значением b в точке a:

$\forall_{PQABRa}(P(f,x) = R(f(a)) \to \text{card}(\text{set}_{fx}(P(f,x)$ & Отображение$(f, A(x), B(x))$ & взаимнооднозначно(f) & $Q(f,x))) = \text{card}(\text{set}_{gbx}(R(b)$ & $b \in B(x)$ & Отображение$(g, A(x)\setminus\{a\}, B(x)\setminus\{b\})$ & взаимнооднозначно(g) & $Q($доопределение$(g, \{a\}, b), x))))$

Указатель "контекст(подчинено(x3 фикс(0 1 1 3))вид(x3 значение(x6 x1))свобоперанд(x1))" идентифицирует элемент a по какому-то вхождению выражения $f(a)$ в условия под описателем. Указатель "перечень(x40 вхождениетерма(x40 значение(x6 x1)))" определяет идентификацию утверждения $P(f,x)$ как конъюнкции всех членов, содержащих выражение $f(a)$. Указатель "новаргумент(x42 x6 фикс)" поясняет, что первый антецедент идентифицирует $P(f,x)$ как $R(f(a))$, проверяя отсутствие вхождений f в ином виде, чем $f(a)$.

Аналогичный прием, где условие взаимной однозначности f отсутствует, имеет следующий вид:

$\forall_{PQABRa}(P(f,x) = R(f(a)) \to \text{card}(\text{set}_{fx}(P(f,x)$ & Отображение$(f, A(x), B(x))$ & $Q(f,x))) = \text{card}(\text{set}_{fgbx}(R(b)$ & $b \in B(x)$ & Отображение$(g, A(x) \setminus \{a\}, B(x))$ & $f =$ доопределение$(g, \{a\}, b)$ & $Q(f,x))))$

Отличие от предыдущего приема состоит в том, что при идентификации $R(f(a))$ допускаются вхождения f вне выражений $f(a)$. Соответственно, вместо указателя "новаргумент(...)" берется указатель "заменатермов(фикс(1 2))".

6. Параметризация пары.

   Если один из перечисляемых параметров есть набор длины 2, то он представляется явным образом как пара новых параметров:

   $\forall_P(\text{card}(\text{set}_{ij}(l(j) = 2$ & $P(i,j))) = \text{card}(\text{set}_{ijk}(P(i,(j,k)))))$

Условие непустоты пересечения с заданным множеством

Для нахождения числа подмножеств x множества $B(y)$, удовлетворяющих некоторому условию $f(x,y)$ и пересекающихся с заданным множеством $A(y)$, из общего числа подмножеств вычитается число подмножеств разности $B(y) \setminus A(y)$:

$\forall_{ABf}(\text{card}(\text{set}_{xy}(\neg($непересек$(x, A(y)))$ & $f(x,y)$ & $x \subseteq B(y))) = \text{card}(\text{set}_{xy}(f(x,y)$ & $x \subseteq B(y))) - \text{card}(\text{set}_{xy}(x \subseteq B(y) \setminus A(y)$ & $f(x,y))))$

Прямое произведение на одноэлементное множество

$\forall_{ab}(\text{card}(\{b\} \times a) = \text{card}(a))$

$\forall_{ab}(\text{card}(a \times \{b\}) = \text{card}(a))$

Подразбиение множества

Если перечисляются подмножества C множества A, условие на которые содержит упоминание о мощности пересечения C с некоторым подмножеством B множества A, то C представляется как объединение подмножества M множества B и подмножества N множества $A \setminus B$:

$\forall_{ABP}(B \subseteq A \to \mathrm{card}(\mathrm{set}_C(C \subseteq A \ \& \ C - \mathrm{set} \ \& \ P(C))) = \mathrm{card}(\mathrm{set}_{MN}(M \subseteq B \ \& \ N \subseteq A \setminus B \ \& \ M - \mathrm{set} \ \& \ N - \mathrm{set} \ \& \ P(M \cup N))))$

Указатель "контекст(позиция(x1 фикс(0 1 1))вид(x1 мощность(пересечение(x27 x28))) контекст(операнд(x2 x1)символ(x2 равно меньшеилиравно)))" дает идентификацию подмножества B по встречающемуся под описателем равенству либо неравенству для $\mathrm{card}(B \cap C)$.

Другой случай подразбиения перечисляемых множеств x заданной мощности m возникает, если x является подмножеством объединения двух непересекающихся множеств $a(y), b(z)$, где параметры y, z независимы друг от друга. Тогда можно представить x как объединение i-элементного подмножества $a(y)$ и $m - i$- элементного подмножества $b(z)$, перемножить количества таких подмножеств, и просуммировать произведения по всем i от 0 до m:

$\forall_{abcPQm}(\text{непересек}(a(y), b(z)) \ \& \ m - \text{натуральное} \to \mathrm{card}(\mathrm{set}_{xyz}(x \subseteq (a(y) \cup b(z)) \setminus c \ \& \ \mathrm{card}(x) = m \ \& \ x - \mathrm{set} \ \& \ P(y) \ \& \ Q(z))) = \sum_{i=0}^{m}(\mathrm{card}(\mathrm{set}_{xy}(x \subseteq a(y) \setminus c \ \& \ \mathrm{card}(x) = i \ \& \ x - \mathrm{set} \ \& \ P(y)))\mathrm{card}(\mathrm{set}_{xz}(x \subseteq b(z) \setminus c \ \& \ \mathrm{card}(x) = m - i \ \& \ x - \mathrm{set} \ \& \ Q(z)))))$

Параметр c введен для обобщения и может принимать вырожденное значение $\emptyset$. Мощности, перемножаемые при суммировании, упрощаются с помощью вспомогательных задач. При обработке первого антецедента проверочным оператором вводятся дополнительные посылки $P(y), Q(z)$.

Последний прием данной серии рассматривает перечисление таких m - элементных подмножеств x множества A, что разность $x \setminus B$ включает множество y (оно также относится к перечисляемым параметрам). Здесь B - некоторое фиксированное подмножество множества A. Прием представляет x как объединение двух множеств u, v; первое является подмножеством B, а второе - подмножеством $A \setminus B$:

$\forall_{ABPm}(B \subseteq A \ \& \ m - \text{натуральное} \to \mathrm{card}(\mathrm{set}_{xyz}(x \subseteq A \ \& \ y \subseteq x \setminus B \ \& \ \mathrm{card}(x) = m \ \& \ P(x, y, z))) = \sum_{i=0}^{m} \mathrm{card}(\mathrm{set}_{uvyz}(u \subseteq B \ \& \ \mathrm{card}(u) = i \ \& \ v \subseteq A \setminus B \ \& \ \mathrm{card}(v) = m - i \ \& \ P(u \cup v, y, z) \ \& \ y \subseteq v \ \& \ u - \mathrm{set} \ \& \ v - \mathrm{set})))$

Число наборов, для которых указано условие на множество их элементов

Пусть перечисляются наборы y заданной длины k, дополнительное условие на y зависит только от множества элементов набора y, а мощность этого множества равна длине набора (т.е. элементы набора различны). Тогда находится число k - элементных подмножеств рассматриваемого множества A, и для получения требуемого количества наборов оно умножается на $k!$:

$\forall_{PAk}(\mathrm{card}(\{; y\}) = k \to \mathrm{card}(\mathrm{set}_y(\text{кортеж}(y, k, A) \ \& \ P(\{; y\}))) = k!\mathrm{card}(\mathrm{set}_y(P(y) \ \& \ y \subseteq A)))$

Первый антецедент обрабатывается вспомогательной задачей на доказательство, которой придается дополнительная посылка $P(\{; y\})$. Указатель "новаргумент(x40 x24 фикс)" обеспечивает отсутствие внутри $P(\ldots)$ вхождений y вне выражений $\{; y\}$.

Множества, содержащие заданный элемент

При нахождении количества подмножеств множества A, имеющих мощность n и содержащих заданный элемент a, можно перейти к рассмотрению $n - 1$-элементных подмножеств множества $A \setminus \{a\}$:

$\forall_{Aanf}(a \in A$ & конечное$(A) \rightarrow \mathrm{card}(\mathrm{set}_{xy}(x - \mathrm{set}$ & $x \subseteq A$ & $a \in x$ & $\mathrm{card}(x) = n$ & $f(x, y))))$

Если усмотреть принадлежность элемента a множеству A не удается, то применяется следующая версия приема:

$\forall_{aPb}(\mathrm{card}(\mathrm{set}_x(a \in x$ & $x - \mathrm{set}$ & $x \subseteq b$ & $P(x))) = \mathrm{card}(\mathrm{set}_x(x - \mathrm{set}$ & $x \subseteq b \setminus \{a\}$ & $P(\{a\} \cup x)$ & $a \in b)))$

Выделение элемента набора, для которого указано особое условие

Пусть перечисляются наборы элементов множества A, для i -го разряда которых предусмотрено специальное условие P. Если выясняется, что число n значений оставшихся разрядов не зависит от выбора значения i - го разряда, то для получения результата оно умножается на число элементов множества A, удовлетворяющих условию P:

$\forall_{APQimn}(m = \mathrm{card}(\mathrm{set}_x(x \in A$ & $P(x)))$ & $n = \mathrm{card}(\mathrm{set}_x($перестановка$(x, A \setminus \{a\})$ & $Q($Вставка$(x, i, a))))$ & конечное$(A) \rightarrow \mathrm{card}(\mathrm{set}_x($перестановка$(x, A)$ & $P(x(i))$ & $Q(x)$ $)) = mn)$

Указатель "контекст(позиция(x2 фикс(0 1 1 2))свобоперанд(x2)вид(x2 значение(x23 x9)))" определяет идентификацию номера i по какому-то вхождению выражения $x(i)$ под описателем "класс". Указатель "перечень(x40 вхождениетерма(x40 значение(x23 x9)))" относит к $P(x(i))$ все члены, содержащие подвыражение $x(i)$. Указатель "новаргумент(x40 x23 фикс)" обеспечивает проверку того, что каждое вхождение переменной x в $P(\ldots)$ расположено внутри выражения $x(i)$. Указатель "новаяпеременная(x1)" означает, что во втором антецеденте в качестве a выбирается некоторая новая переменная, а фильтр "не(входит(x1 x14))" проверяет независимость от a найденной мощности n. Для вычисления мощностей m, n используются вспомогательные задачи. Создана еще одна версия данного приема, в которой вместо символа "перестановка" используется символ "кортеж":

$\forall_{APQimn}(m = \mathrm{card}(\mathrm{set}_x(x \in A$ & $P(x)))$ & $n = \mathrm{card}(\mathrm{set}_x($кортеж$(x, k-1, A)$ & $Q($Вставка$(x, i, a))))$ & конечное$(A) \rightarrow \mathrm{card}(\mathrm{set}_x($кортеж$(x, k, A)$ & $P(x(i))$ & $Q(x))) = mn)$

Переход от двоичного набора к множеству номеров позиций, на которых стоят единицы

$\forall_{nA}(\mathrm{card}(\mathrm{set}_{xy}($двнабор$(x, n)$ & $A($слой$(x, 1), y))) = \mathrm{card}(\mathrm{set}_{xy}(x \subseteq \{1, \ldots, n\}$ & $A(x, y))))$

Упрощение аргумента функциональной переменной

Если перечисляемый параметр x встречается в выражении вида $f(a \pm x)$, где f - переменная, а отличных от $a \pm x$ аргументов функциональных переменных нет, то выполняется линейное преобразование для перехода к выражению $f(x)$:

$\forall_{Afab}(\mathrm{card}(\mathrm{set}_{xy}(A(x, y, f(a + bx)))) = \mathrm{card}(\mathrm{set}_{xy}(A((x - a)/b, y, f(x)))))$

Коэффициент b принимает значения ± 1, что обеспечивается указателем "или(заголовок(x2 1)и(заголовок(x2 минус)первыйсимвол(x2 1)))". Фильтр "не(контекст(позиция(x3 фикс(0 1 1 3))вид(x3 значение(x7 x8))входит(x23 x8)не(равно(x8 фикс(0 1 1 3 2 3 2)))))" проверяет отсутствие других содержащих x аргументов функциональных переменных.

Перестановка переменных связывающей приставки

В специальных случаях перестановка переменных связывающей приставки может использоваться для дополнительной стандартизации условий под описателем:

$\forall_P(\mathrm{card}(\mathrm{set}_{xy}(y+1 \leq x \ \& \ P(x,y))) = \mathrm{card}(\mathrm{set}_{yx}(y+1 \leq x \ \& \ P(x,y))))$

$\forall_{Pn}(\mathrm{card}(\mathrm{set}_{ij}(i \in \{1,\ldots,n\} \ \& \ j \in \{1,\ldots,i-1\} \ \& \ P(i,j))) = \mathrm{card}(\mathrm{set}_{ji}(j \in \{1,\ldots,n\} \ \& \ i \in \{j+1,\ldots,n\} \ \& \ P(i,j))))$

Мощность образа при взаимно-однозначном отображении

$\forall_{fA}(A \subseteq \mathrm{Dom}(f) \ \& \ \text{взаимнооднозначно}(f) \to \mathrm{card}(\text{образ}(f,A)) = \mathrm{card}(A))$

6.6 Суммы и разности мощностей множеств

Мощность объединения непересекающихся множеств

Если удается усмотреть, что два множества не пересекаются, то мощность их объединения заменяется на сумму мощностей:

$\forall_{ab}(\text{непересек}(a,b) \to \mathrm{card}(a \cup b) = \mathrm{card}(a) + \mathrm{card}(b))$

Прием имеет фильтр, допускающий либо числовые значения мощности (случай конечных множеств), либо такие случаи, в которых подстановка бесконечных значений мощности не приводит к коллизиям. Кроме приема замены, для мощности объединения непересекающихся множеств создан также прием вывода в посылках:

$\forall_{abc}(a = b \cup c \ \& \ \text{непересек}(b,c) \to \mathrm{card}(a) = \mathrm{card}(b) + \mathrm{card}(c))$

Указатель "контрольвывода(мощность(x1))" определяет инициализацию приема при усмотрении выражения $\mathrm{card}(a)$. Первый антецедент идентифицируется с некоторой посылкой и позволяет определить множества b, c. После этого выводится следствие. Для блокировки повторных срабатываний используется комментарий (мощность x1).

Мощность разности вложенных множеств

Прежде всего, имеются два приема, аналогичных приемам предыдущего пункта:

$\forall_{ab}(a \subseteq b \ \& \ \text{конечное}(b) \to \mathrm{card}(b \setminus a) = \mathrm{card}(b) - \mathrm{card}(a))$

$\forall_{abc}(b = a \setminus c \ \& \ c \subseteq a \ \& \ \text{конечное}(a) \to \mathrm{card}(a) = \mathrm{card}(b) + \mathrm{card}(c))$

Первый прием выполняет тождественное преобразование, второй - вывод следствия в посылках. Он инициируется при усмотрении выражения $\mathrm{card}(a)$; первый антецедент идентифицируется с посылкой, второй и третий - обрабатываются проверочными операторами.

Дальнейшие приемы преобразуют линейную комбинацию двух мощностей вложенных множеств, у которой коэффициенты имеют разный знак, а множества задаются описателями:

1. $\forall_{ABmn}(mn < 0 \to m\mathrm{card}(\mathrm{set}_x(A(x) \ \& \ B(x))) + n\mathrm{card}(\mathrm{set}_y(A(y))) = n\mathrm{card}(\mathrm{set}_x(A(x) \ \& \ \neg(B(x)))) + (m+n)\mathrm{card}(\mathrm{set}_x(A(x) \ \& \ B(x))))$

   Коэффициенты m, n идентифицируются с целыми числами; функциональная переменная $B(x)$ - с элементарным утверждением.

2. $\forall_{ABmnC}(mn < 0 \ \& \ B(x) \ \& \ (B(x) \ \& \ \neg(A(x))) = C(x) \to m\mathrm{card}(\mathrm{set}_x(A(x))) + n\mathrm{card}(\mathrm{set}_y(B(y))) = n\mathrm{card}(\mathrm{set}_x(C(x))) + (m+n)\mathrm{card}(\mathrm{set}_x(A(x))))$

   Как и в предыдущем случае, коэффициенты m, n идентифицируются с целыми числами. Второй антецедент обрабатывается вспомогательной задачей на доказательство, которой придается дополнительная посылка $A(x)$. Таким образом устанавливается вложенность множеств. Третий антецедент выделен указателем "идентификатор". Он определяет условие $C(x)$ принадлежности разности множеств, получаемое при решении задачи на преобразование. Проверяется, что $C(x)$ не является дизъюнкцией.

3. $\forall_{mnPABCD}(m + n = 0 \to m\mathrm{card}(\mathrm{set}_{xy}(P(x,y) \ \& \ x \in A \setminus B \ \& \ y \in C(x) \setminus D(x))) + n\mathrm{card}(\mathrm{set}_{uv}(P(u,v) \ \& \ u \in A \ \& \ v \in C(u))) = n\mathrm{card}(\mathrm{set}_{xy}(P(x,y) \ \& \ x \in B \ \& \ y \in C(x) \setminus D(x))) + n\mathrm{card}(\mathrm{set}_{xy}(P(x,y) \ \& \ x \in A \ \& \ y \in C(x) \cap D(x))))$

   Первый антецедент в этом и следующем приемах выделен указателем "идентификатор".

4. $\forall_{mnPAB}(m+n = 0 \ \& \ A \subseteq B \to m\mathrm{card}(\mathrm{set}_x(P(x) \ \& \ x \in A)) + \mathrm{card}(\mathrm{set}_y(P(y) \ \& \ y \in B)) = n\mathrm{card}(\mathrm{set}_y(P(y) \ \& \ y \in B \setminus A)))$

Дизъюнктивное условие под описателем

Если конъюнктивный член условия принадлежности классу имеет заголовком дизъюнкцию, то в случае несовместных подслучаев мощности подмножеств складываются:

$\forall_{fgh}(\forall_x(h(x) \ \& \ f(x) \to \neg(g(x))) \to \mathrm{card}(\mathrm{set}_x((f(x) \vee g(x)) \ \& \ h(x))) = \mathrm{card}(\mathrm{set}_x(f(x) \ \& \ h(x))) + \mathrm{card}(\mathrm{set}_x(g(x) \ \& \ h(x))))$

Указатель "легковидеть(1)" определяет обработку первого антецедента, проверяющего несовместность подслучаев, с помощью вспомогательной задачи на доказательство. Уровень обращения к ней равен 4. Ограничитель "лимит(1900000)" отводит для проверки достаточно большой ресурс. Если подслучаи не обязательно взаимно исключающие, то используется другой прием, предварительно вычисляющий мощность пересечения:

$\forall_{PQRa}(a = \mathrm{card}(\mathrm{set}_x(P(x) \ \& \ Q(x) \ \& \ R(x))) \ \& \ a - \text{число} \to \mathrm{card}(\mathrm{set}_x((P(x) \vee Q(x)) \ \& \ R(x))) = \mathrm{card}(\mathrm{set}_x(P(x) \ \& \ R(x))) + \mathrm{card}(\mathrm{set}_x(Q(x) \ \& \ R(x))) - a)$

Первый антецедент, выделенный указателем "идентификатор", вычисляет мощность пересечения подмножеств, соответствующих подслучаям. Его правая часть обрабатывается вспомогательной задачей на преобразование. Второй антецедент проверяет конечность найденной мощности. Фильтр "не(заголовок(x1 мощность))" проверяет успех попытки вычисления.

Наконец, добавлен прием, в котором дизъюнкция неявная - имеет вид принадлежности параметра двухэлементному множеству:

$\forall_{abP}(\neg(a(y) = b(y)) \to \mathrm{card}(\mathrm{set}_{xy}(x \in \{a(y), b(y)\} \ \& \ P(x,y))) = \mathrm{card}(\mathrm{set}_y(P(a(y), y))) + \mathrm{card}(\mathrm{set}_y(P(b(y), y))))$

Перечисление отображений, у которых мощность прообраза заданного элемента строго меньше мощности области определения

Мощность семейства отображений представляется как разность мощности семейства отображений с отброшенным условием на прообраз, и мощности семейства константных отображений:

$\forall_{ABCbp}(\mathrm{card}(A(x)) - p = 1 \to \mathrm{card}(\mathrm{set}_{fx}(\text{Отображение}(f, A(x), B(x)) \ \&\ \mathrm{card}(\text{слой}(f, b)) \leq p \ \&\ C(f, x))) = \mathrm{card}(\mathrm{set}_{fx}(\text{Отображение}(f, A(x), B(x)) \ \&\ C(f, x))) - \mathrm{card}(\mathrm{set}_{fx}(f = \text{конст}(A(x), b) \ \&\ C(f, x))))$

Первый антецедент выделен указателем "идентификатор". Мощности в заменяющей разности вычисляются с помощью вспомогательных задач на преобразование.

Переход от суммы мощностей к мощности множества

Приемы, вычисляющие мощность путем суммирования по значениям параметров, вводили комментарий "разбиение". Если этого комментария нет, то выполняется обратное преобразование:

$\forall_{fg}(\sum_{a,f(a)} \mathrm{card}(\mathrm{set}_b(g(a, b))) = \mathrm{card}(\mathrm{set}_{ab}(f(a) \ \&\ g(a, b))))$

Мощность множества объектов, имеющих нечетную целочисленную характеристику

Происходит суммирование значений данной характеристики по модулю 2:

$\forall_{fp}(\mathrm{card}(\mathrm{set}_x(f(x) \ \&\ \neg(p(x) - \mathrm{even}))) = \sum_{x,f(x)}(p(x)(\mathrm{mod}2)))$

Равенство суммы мощностей двух множеств мощности третьего

Если множество C содержится в объединении множеств A, B, то равенство мощности C сумме мощностей множеств A, B означает, что эти множества не пересекаются и содержатся в C:

$\forall_{ABC}(C \subseteq A \cup B \to \mathrm{card}A + \mathrm{card}B = \mathrm{card}C \leftrightarrow A \subseteq C \ \&\ B \subseteq C \ \&\ A \cap B = \emptyset)$

Выделение подмножества, определяемого конечным промежутком значений числовой характеристики

Если вычитаются два множества, определяемых нижними границами для значений некоторой числовой характеристики, то вводится множество для соответствующего конечного промежутка значений данной характеристики:

$\forall_{ABabmn}(m + n = 0 \ \&\ A \subseteq B \to m\mathrm{card}(\mathrm{set}_x(a < p(x) \ \&\ x \in A)) + n\mathrm{card}(\mathrm{set}_y(b < p(y) \ \&\ y \in B)) = n\mathrm{card}(\mathrm{set}_x(b < p(x) \ \&\ p(x) \leq a \ \&\ x \in A)) + n\mathrm{card}(\mathrm{set}_x(b < p(x) \ \&\ x \in B \setminus A)))$

Первый антецедент выделен указателем "идентификатор"; второй - "блокпроверок". Указатели "альтернатива" учитывают возможность замены любого из строгих неравенств на нестрогое. Для случая, когда вычитаются множества, определяемые верхними границами характеристики p, введен аналогичный прием.

Усмотрение дополняющих друг друга множеств

Если в одном и том же терме задачи рассматриваются мощности множеств, определяемых условиями $P(x)\&R(x)$, $P(x)\&Q(x)$, причем члены $R(x), Q(x)$ несовместны, а дизъюнкция их эквивалентна элементарному утверждению $S(x)$, то предпринимается попытка вычислить мощность n объединения, определяемого условиями $P(x), S(x)$. Затем мощность первого множества выражается как вычитание из n мощности второго:

$\forall_{PQRSn}(\neg R(x)\ \&\ (R(x)\vee Q(x)) = S(x)\ \&\ \mathrm{card}(\mathrm{set}_x(P(x)\ \&\ S(x))) = n\ \&\ A = \mathrm{set}_y(P(y)\ \&\ R(y)) \rightarrow \mathrm{card}(\mathrm{set}_x(P(x)\ \&\ Q(x))) = n - \mathrm{card}(\mathrm{set}_y(P(y)\ \&\ R(y))))$

Указатель "контекст(позиция(x2 корень)вид(x2 мощность(x26)))" определяет идентификацию внутри текущего терма задачи выражения $\mathrm{card}A$. Четвертый антецедент выделен указателем "идентификатор", так что A далее идентифицируется с выражением $\mathrm{set}_y(P(y)\ \&\ R(y))$. Первый антецедент выделен указателем "легковидеть". Он проверяет несовместность условий $R(x), Q(x)$; для проверки используется вспомогательная задача, посылки которой пополнены утверждениями $P(x), Q(x)$. Второй антецедент выделен указателем "идентификатор". Его левая часть упрощается вспомогательной задачей. Фильтр "элементарно(значение(x43 x23))" проверяет, что результат упрощения - элементарное утверждение. Наконец, третий антецедент, тоже выделенный указателем "идентификатор", вычисляет мощность n. Используется фильтр "не(входит(мощность x14))". Во избежание попыток применять прием к вхождениям одного и того же описателя "класс" добавлен ускоряющий фильтр "не(равно(значение(x42 x43)значение(x41 x43)))". Фильтр "входит(значение фикс(0 1 1))" ограничивает применение приема только теми ситуациями, когда под описателем имеется функциональная переменная.

Кроме приема замены, добавлен аналогичный прием вывода, который идентифицирует оба множества из посылок $A = \mathrm{card}(\mathrm{set}_y(P(y)\ \&\ R(y)))$, $B = \mathrm{card}(\mathrm{set}_x(P(x)\ \&\ Q(x)))$.

6.7 Неравенства для мощности

Оценка мощности подмножества

Прием усматривает истинность неравенства, утверждающего, что мощность подмножества не превосходит мощности множества:

$\forall_{ab}(a \subseteq b\ \&\ c = \mathrm{card}b \rightarrow \mathrm{card}a \leq c)$

Включение в объединение

Чтобы доказать, что мощность множества не превосходит суммы мощностей некоторых других множеств, предпринимается попытка усмотреть включение его в объединение этих множеств:

$\forall_{ABn}(A \subseteq \bigcup_{i=1}^{n} B(i) \rightarrow 0 \leq \sum_{i=1}^{n} \mathrm{card}B(i) - \mathrm{card}A)$

Прием срабатывает на уровне 7. Он применяется к неравенству, представляющему собой условие задачи на доказательство, и заменяет его на константу "истина". Антецедент обрабатывается вспомогательной задачей на доказательство. Указатель "развертка(фикс(1 2)фикс(0 2 1))" определяет идентификацию конечной суммы с обычной суммой нескольких слагаемых, а также запись объединения семейства как обычного объединения нескольких множеств.

Бесконечное множество

Прием усматривает истинность неравенства, в левой части которого расположено число, а в правой - мощность бесконечного множества:

$\forall_{aA}(a - \text{число} \ \& \ \neg(\text{конечное}(A)) \to a \leq \text{card}A)$

Разбор случаев для мощности множества, упоминаемого в кванторной импликации

Если имеется кванторная импликация, которая для любых двух различных элементов x, y множества A гарантирует выполнение отношения $P(x, y)$, причем большое число условий задачи на описание имеют вид $P(X, Y)$, где X, Y - неизвестные, то полезным оказывается разбор случаев на мощность множества A. Если эта мощность достаточно велика, то имеется необходимое число пар различных элементов в A для реализации указанных условий $P(X, Y)$. Соответственно, задача будет сведена к рассмотрению подслучая для малой мощности множества A. Теорема приема имеет вид:

$\forall_{APBkm}(A \subseteq B \ \& \ \text{card}B = m \ \& \ \forall_{xy}(x \in A \ \& \ y \in A \ \& \ \neg(x = y) \to P(x, y)) \to k \leq \text{card}A \vee \text{card}A < k)$

Указатель "контекст(условие(x1)вид(x1 принадлежит(x2 x27))неизвестная(x2)известно(x27))" идентифицирует известное множество B, для которого имеется условие принадлежности ему некоторой неизвестной. Третий антецедент идентифицируется с посылкой задачи на описание, причем в нем же выбирается точка привязки. Первый антецедент обрабатывается проверочным оператором; второй - выделен указателем "идентификатор". Он определяет мощность m множества B; фильтр "известно(x13)" проверяет, что эта мощность не содержит неизвестных. Указатель "контекст(равно(x11 число(неизвестная(x3)контекст(условие(x4)вид(x4 принадлежит(x3 x27)))контекст(условие(x5)входит(x3 x5)вид(x5 значение(x40 набор(x6 x7)))неизвестная(x6)неизвестная(x7)))))" определяет число k таких неизвестных задачи, для которых имеется условие их принадлежности множеству B, а также условие вида $P(X, Y)$, содержащее данную неизвестную. Прием имеет заголовок "вывод". Указатель "примечание(разборслучаев)" форсирует использование выведенной дизъюнкции для разбора случаев. Заметим, что поводом для создания приема послужила известная задача "среди любых шести человек найдутся либо трое попарно знакомых, либо трое попарно незнакомых", иллюстрирующая теорему Рамсея. Ее можно найти в разделе задачника "Теория графов".

Усмотрение противоречия из неположительности мощности

Если множество непусто, то условие неположительности его мощности противоречиво:

$\forall_{Ab}(\text{card}A = b \ \& \ \neg(A = \emptyset) \ \& \ b \leq 0 \to \text{ложь})$

Прием применяется в задачах на исследование, имеющих цель "контроль", т.е. после разбора случаев.

Усмотрение верхней оценки мощности прообраза элемента

Для усмотрения истинности верхней оценки мощности прообраза элемента достаточно установить, что она не менее мощности области определения:

$\forall_{ABbp}(b \in B \ \& \ \text{Отображение}(f, A, B) \ \& \ 0 \leq p - \text{card}A \to \text{card}(\text{слой}(f, b)) \leq p)$

6.8 Выделение независимого условия на принадлежность множеству константной мощности

Если мощность класса допустимых значений одного из перечисляемых параметров не зависит от оставшихся параметров, то она выносится в качестве множителя:

$\forall_{APb}(\text{card}A(y) = b \to \text{card}(\text{set}_{xy}(x \in A(y) \ \& \ P(y))) = b\text{card}(\text{set}_y(P(y))))$

Первый антецедент выделен указателем "идентификатор". Он вычисляет мощность множества $A(y)$ с помощью вспомогательной задачи на преобразование. Фильтр "не(пересекаются(x24 x2))" проверяет независимость найденной мощности от параметров y.

6.9 Усмотрение разбиения на непересекающиеся подмножества

Пусть вычисляется мощность множества, заданного таким условием, внутри которого встречается равенство некоторого множества D объединению множеств $A(i); i = 1, \ldots, n$. Если оказывается, что сумма мощностей множеств $A(i)$ равна мощности множества D, т.е. $A(i)$ не пересекаются, то равенство для D заменяется на конъюнкцию включений каждого $A(i)$ в разность D и объединения остальных $A(j)$:

$\forall_{ADnm}(\forall_i(i \in \{1, \ldots, n\} \to \text{card}A(i) = m(i)) \ \& \ \sum_{i=1}^n m(i) - \text{card}D = 0 \to D = \bigcup_{i=1}^n A(i) \leftrightarrow \forall_i(i \in \{1, \ldots, n\} \to A(i) \subseteq D \setminus \bigcup_{j=1}^{i-1} A(j)))$

Указатель "развертка(фикс(1)фикс(2 1 1)фикс(0 1 2)фикс(0 2)фикс(0 2 5 2 2))" определяет идентификацию объединения семейства с обычным объединением нескольких множеств, а также соответствующую интерпретацию остальных связанных с множествами $A(i)$ фрагментов теоремы. Фильтр "контекст(подчинено(теквхожд (x1)символ(x1 мощность))" проверяет, что преобразуемое приемом равенство находится внутри выражения "мощность(...)". Первый антецедент дает конъюнкцию непосредственно идентифицируемых равенств для мощности; второй - выделен указателем "идентификатор".

6.10 Стандартизация равенств и неравенств с мощностью

Использование оператора "верхняяоценка" для усмотрения ложности равенства

$\forall_{amn}(\text{card}a \leq m \ \& \ 0 < n - m \to \neg(\text{card}a = n))$

Первый антецедент выделен указателем "значения". Он обращается к оператору "верхняяоценка" для нахождения верхней оценки m мощности множества a. Второй антецедент, выделенный указателем "блокпроверок", устанавливает, что эта мощность меньше числа n. Прием имеет заголовок "второйтерм" и заменяет равенство на константу "ложь". Фильтры "целое(x13)", "целое(x14)" означают, что переменные m, n идентифицируются с целочисленными константами.

Равенство мощности нулю

$\forall_a(a - \mathrm{set} \to \mathrm{card}a = 0 \leftrightarrow a = \emptyset)$

Равенство мощности конечного списка его длине

Если в условии задачи на описание встречается равенство мощности конечного списка его длине, то это условие переформулируется в терминах попарного различия элементов списка. Рекурсивный шаг переформулировки реализуется следующим приемом:

$\forall_{abmn}(l(b) = n \ \& \ m = n + 1 \to \mathrm{card}\{a; b\} = m \leftrightarrow \neg(a \in \{; b\}) \ \& \ \mathrm{card}\{; b\} = n)$

Фильтр "натуральное(x13)" означает, что m идентифицируется с натуральной константой. Первый антецедент вычисляет длину n остаточной части b; второй - сравнивает длину и мощность списка. Фильтр "символ(фикс(0 1 1 1 1)набор)" проверяет, что список задан явным перечислением элементов. При выборе a предпочтение отдается неконстантным элементам.

Выражение мощности из линейного уравнения

Если встречается линейное относительно мощности уравнение, то оно разрешается относительно мощности:

$$\forall_{abcA}(\neg(a = 0) \to a\mathrm{card}A + b = c \leftrightarrow \mathrm{card}A = \frac{c - b}{a}$$

Фильтры проверяют отсутствие символа "мощность" в выражениях a, b, c. Предусмотрены две версии приема: одна для случая, когда равенство представляет собой посылку, другая - для случая, когда оно входит под описателем "класс", причем A является одним из параметров описателя.

Исключение условного выражения

При обучении решателя возникли две частных ситуации, потребовавших специальных приемов исключения условного выражения в равенстве для мощности. Первый прием усматривает нереализуемость одной из альтернатив:

$\forall_{abcPd}(0 < d - l(c) - 1 \to \mathrm{card}\{a; (b \text{ при } P, \text{иначе } c)\} = d \leftrightarrow \mathrm{card}\{a; b\} = d \ \& \ P)$

Фильтры "заголовок(x3 набор)", "натуральное(x4)" проверяют, что выражение c явно перечисляет некоторый набор, причем d идентифицируется с натуральной константой. Антецедент убеждается, что длина набора меньше $d - 1$. Второй прием просто реализует разбор случаев:

$\forall_{Apqm}(\mathrm{card}((p \text{ при } A, \ \text{иначе } q)) = m \leftrightarrow A \ \& \ \mathrm{card}p = m \vee \neg A \ \& \ \mathrm{card}q = m)$

Мощность множества корней квадратного трехчлена

Выше был приведен прием, дающий выражение для мощности множества корней квадратного уравнения. Это выражение достаточно громоздкое, и в случаях равенства мощности единице либо двойке можно сразу выписать необходимый его фрагмент:

$\forall_{abc}(\mathrm{card}(\mathrm{set}_x(x - \text{число} \ \& \ ax^2 + bx + c = 0)) = 1 \leftrightarrow \neg(a = 0) \ \& \ b^2 - 4ac = 0 \vee a = 0 \ \& \ \neg(b = 0))$

$\forall_{abc}(\mathrm{card}(\mathrm{set}_x(x - \text{число} \ \& \ ax^2 + bx + c = 0)) = 2 \leftrightarrow 0 < b^2 - 4ac)$

Уровень срабатывания приемов равен 0; уровень срабатывания приема, дающего выражение мощности в общем случае, был равен 1.

Мощность конечного списка не больше его длины

Прием заменяет на логическую константу "ложь" нижнюю оценку мощности списка, превосходящую его длину:

$\forall_{amn}(0 < m - n \ \& \ l(a) = n \rightarrow \neg(m \leq \mathrm{card}\{;a\}))$

Мощность объединения двух множеств, каждое из которых не более чем одноэлементное

Если для мощности объединения двух не более чем одноэлементных множеств имеется нижняя оценка 2, то она заменяется на условия одноэлементности каждого множества и различия этих элементов:

$\forall_{ABab}(2 \leq \mathrm{card}((\{a\} \text{ при } A, \text{ иначе } \emptyset) \cup (\{b\} \text{ при } B, \text{ иначе } \emptyset)) \leftrightarrow \neg(a = b) \ \& \ A \ \& \ B)$

Ориентация равенства с мощностью

Равенства вида $a = \mathrm{card}(b)$, встречающиеся в посылках либо в условии задачи на описание, преобразуются к виду $\mathrm{card}(b) = a$. При этом они снабжаются комментарием "ориентацияравенства", блокирующим обратную перестановку членов. Такая ориентация равенств способствует подстановке вместо мощностей множеств их явных выражений.

Мощность трехэлементного списка

Мощность трехэлементного списка не превосходит 2 тогда и только тогда, когда какие-то два элемента совпадают:

$\forall_{abc}(\mathrm{card}\{a, b, c\} \leq 2 \leftrightarrow a = b \vee a = c \vee b = c)$

Прием применяется к неравенству, являющемуся содержащим неизвестные условием задачи на описание.

Оценка мощности пересечения

Если p - элементное множество A имеет в пересечении с множеством B не менее чем m элементов, причем B не пересекается с множеством C, то на долю пересечения A и C остается не более $p - m$ элементов:

$\forall_{ABCmnp}(m \le \mathrm{card}(A \cap B)\ \&\ \text{непересек}(B, C)\ \&\ \mathrm{card}A = p\ \&\ 0 < m+n-p\ \&\ p-mbox \to \neg(n \le \mathrm{card}(A \cap C)))$

Прием имеет заголовок "второйтерм", т.е. заменяет неравенство для мощности множества $A \cap C$ на константу "ложь". Третий антецедент выделен указателем "идентификатор" - он вычисляет мощность p. Второй, четвертый и пятый антецеденты обрабатываются проверочными операторами. Первый антецедент идентифицируется непосредственно - это позволяет определить множество B.

Неотрицательность мощности

Равенство мощности отрицательному выражению заменяется на константу "ложь":

$\forall_{Ab}(b < 0 \to \mathrm{card}A = b \leftrightarrow \text{ложь})$

Выражение одной мощности через другую из линейного соотношения

Рассматривается только соотношение для суммы мощностей:

$\forall_{abc}(c - \text{число} \to \mathrm{card}a + \mathrm{card}b = c \leftrightarrow \mathrm{card}a = c - \mathrm{card}b)$

К числу дополнительных ограничений на срабатывание приема относится требование существования внешнего описателя "класс", в связывающую приставку которого входят переменные a, b.

Усмотрение эквивалентных условий принадлежности

Если требуется доказать равенство мощностей двух множеств, заданных описателями "класс", то предпринимается попытка усмотреть эквивалентность условий под описателями:

$\forall_{PQ}(\forall_{xy}(P(x, y) \to Q(x, y))\ \&\ \forall_{xy}(Q(x, y) \to P(x, y)) \to \mathrm{card}(\mathrm{set}_{xy}(P(x, y))) = \mathrm{card}(\mathrm{set}_{zv}(Q(z, v))))$

Прием выполняет не тождественное, а эквивалентное преобразование: он заменяет равенство указанного в консеквенте вида на логическую константу "истина". Это определяется указателем "эквивалентно". Фильтры "условие", "тип(доказать)" требуют, чтобы преобразуемое равенство было условием задачи на доказательство. Указатели "элемент(x25)", "элемент(x21)" разрешают идентификацию переменных связывающей приставки z, v в произвольном порядке - для сравнения мощностей порядок несущественен. Первые два антецедента обрабатываются вспомогательными задачами на доказательство. Уровень срабатывания приема равен 5.

Число подмножеств заданной мощности

Если не удается усмотреть конечность некоторого множества, то для числа его подмножеств заданной мощности невозможно непосредственно применить формулу с числом сочетаний. Однако, если в этой ситуации рассматривается равенство указанного числа подмножеств некоторому числу, то данная формула может быть использована:

$\forall_{abn}(n - \text{целое}\ \&\ 0 \le n\ \&\ b - \text{число}\ \&\ 0 < b \to \mathrm{card}(\mathrm{set}_x(x - \mathrm{set}\ \&\ x \subseteq a\ \&\ \mathrm{card}(x) = n)) = b \leftrightarrow C^n_{\mathrm{card}(a)} = b\ \&\ \mathrm{card}(a) - \text{число}\ \&\ 0 \le \mathrm{card}(a) - n)$

6.11 Существование множества заданной мощности

Чтобы исключать несущественную неизвестную задачи на описание - множество a, для которого фиксировано значение мощности p - созданы следующие два приема:

$\forall_{pb}(p - \text{число} \rightarrow \exists_a(a - \text{set} \ \& \ \text{card}(a) = p \ \& \ \text{непересек}(a, b)) \leftrightarrow p - \text{целое} \ \& \ 0 \leq p)$

$\forall_p(p - \text{число} \rightarrow \exists_a(a - \text{set} \ \& \ \text{card}(a) = p) \leftrightarrow p - \text{целое} \ \& \ 0 \leq p)$

Оба они имеют заголовок "связка". Приемы срабатывают, если задача не имеет других условий с неизвестной a, кроме утверждений $a - \text{set}$, $\text{card}(a) = p$ и (в первом случае) непересек(a, b). Вместо этих условий в задачу помещается требование неотрицательности целочисленного параметра p.

6.12 Существование подмножества заданной мощности

Для исключения несущественной неизвестной задачи на описание - подмножества a заданного конечного множества A, имеющего заданную мощность, создан следующий прием, имеющий заголовок "связка":

$\forall_{Am}(\text{конечное}(A) \rightarrow \exists_x(x \subseteq A \ \& \ \text{card}(x) = m \ \& \ x - \text{set}) \leftrightarrow m \leq \text{card}(A) \ \& \ 0 \leq m)$

6.13 Использование кванторного тождества из посылок

Чтобы найти мощность множества S, можно использовать кванторную импликацию, дающую явное выражение мощности множеств заданного вида $Q(i)$:

$\forall_{PQSaj}(P(j) \ \& \ S = Q(j) \ \& \ \forall_i(P(i) \rightarrow \text{card}Q(i) = a(i)) \rightarrow \text{card}S = a(j))$

Второй антецедент, выделенный указателем "идентификатор", представляет множество S в виде $Q(j)$. Первый антецедент проверяет истинность условия $P(j)$, используя для этого вспомогательную задачу на доказательство. Фильтр "не(входит(мощность терм(значение(x1 x9))))" проверяет, что выражение $a(i)$ не содержит символа "мощность", т.е. может рассматриваться как явное значение мощности.

6.14 Мощность множества, заданного условным выражением

Если множество задано условным выражением, то находятся мощности альтернативных множеств, и из них строится условное выражение для мощности множества:

$\forall_{abcpq}(\text{card}(b) = p \ \& \ \text{card}(c) = q \rightarrow \text{card}((b \text{ при } a, \text{ иначе } c)) = (p \text{ при } a, \text{ иначе } q))$

Первые два антецедента, выделенные указателем "идентификатор", вычисляют мощности альтернативных множеств b, c с помощью вспомогательных задач на преобразование.

6.15 Кванторные условия с трехэлементными подмножествами

Условия принадлежности трехэлементному множеству, встречающиеся под кванторами, переформулируются в терминах данных трех элементов:

$\forall_{AB}(\forall_x(\text{card}(x) = 3 \ \& \ x \subseteq A \ \& \ x - \text{set} \to \exists_{yz}(y \in x \ \& \ z \in x \ \& \ \neg(y = z) \ \& \ B(y,z))) \leftrightarrow \forall_{xyz}(x \in A \,\&\, y \in A \,\&\, z \in A \,\&\, \neg(x = y) \,\&\, \neg(y = z) \,\&\, \neg(x = z) \,\&\, \neg(B(x,y)) \,\&\, \neg(B(x,z)) \to B(y,z)))$

$\forall_{xAB}(x \subseteq A \ \& \ \text{card}(x) = 3 \ \& \ \forall_{yz}(y \in x \ \& \ z \in x \ \& \ \neg(y = z) \to B(y,z)) \leftrightarrow \exists_{uvw}(u \in A \ \& \ v \in A \ \& \ w \in A \ \& \ \neg(u = v) \ \& \ \neg(u = w) \ \& \ \neg(v = w) \ \& \ x = \{u,v,w\} \ \& \ B(u,v) \ \& \ B(u,w) \ \& \ B(v,w)))$

6.16 Условное выражение под описателем

Для исключения условного выражения класс подразбивается на два непересекающихся подкласса, мощности которых суммируются:

$\forall_{Pabq}(\text{card}(\text{set}_x(P(x, (a(x) \text{ при } q(x), \text{ иначе } b(x))))) = \text{card}(\text{set}_x(P(x, a(x)) \ \& \ q(x))) + \text{card}(\text{set}_x(P(x, b(x)) \ \& \ \neg(q(x)))))$

Указатель "вхождение(x40)" определяет идентификацию условного выражения с произвольным подтермом описателя "класс", имеющим заголовок "вариант".

6.17 Нормализатор "нормощность"

В нормализаторе собран ряд перечисленных выше приемов, позволяющих определять мощность множеств в простейших случаях (пустое множество; одноэлементое множество; конечный список; конечный отрезок целых чисел; множества целых, натуральных либо рациональных чисел; объединение непереспкающихся множеств; разность множеств, и т.п.).

6.18 Конечные множества

В дополнение к приемам определения мощности множеств, созданы приемы для усмотрения конечности множеств или преобразований условий конечности.

Конечность разности

Если вычитаемое множество конечное, то условие конечности разности эквивалентно условию конечности того множества, из которого вычитают:

$\forall_{ab}(\text{конечное}(a) \to \text{конечное}(b \setminus a) \leftrightarrow \text{конечное}(b))$

Усмотрение конечности множества из конечности семейства его подмножеств заданной мощности

$\forall_{an}(\text{card}(\text{set}_x(x - \text{set} \ \& \ x \subseteq a \ \& \ \text{card}(x) = n)) - \text{число} \to \text{конечное}(a))$

Прием имеет заголовок "вывод".

Условие конечности числового промежутка

$\forall_{abcd}$(конечное($[a, b]$) $\leftrightarrow b \leq a$)

Переменные c, d обозначают типы концов промежутка, т.е. выражение $[a, b]$ в скобочной записи имеет вид "промежуток($a\ b\ c\ d$)".

Отбрасывание в условии конечности класса избыточного дизъюнктивного члена

Если дизъюнктивный член описания класса фиксирует значение элемента, то он не влияет на конечность класса:

$\forall_{ABCt}$(конечное($\mathrm{set}_x((x = t$ & $A \vee B(x))$ & $C(x))$) $\leftrightarrow$ конечное($\mathrm{set}_x(B(x)$ & $C(x))$))

Отбрасывание альтернативы в условном выражении

Если одно из альтернативных множеств условного выражения конечно, то условие бесконечности относится ко второму множеству, причем фиксируется истинностное значение условия:

$\forall_{Apq}$(конечное(q) $\rightarrow \neg$(конечное((p при A, иначе q))) $\leftrightarrow A$ & $\neg$(конечное(p)))

Проверочный оператор "усмконечное"

Для усмотрения конечности множеств создан проверочный оператор "усмконечное". Обращение к нему выполняется следующим приемом:

$\forall_a$(конечное(a) $\rightarrow$ конечное(a))

Прием имеет заголовок "второйтерм". Его применение к посылкам и условиям вида "конечное(a)", где a - переменная, блокируется. Ограничитель "лимит(20000)" делает попытку проверки очень кратковременной. Проверочный оператор имеет следующие приемы:

1. Пустое множество.

   Теорема приема - "конечное($\emptyset$)".

2. Конечный список.

   $\forall_a$(конечное$\{; a\}$)

3. Конечный отрезок целых чисел.

   $\forall_{mn}$(m – целое & n – целое $\rightarrow$ конечное($\{m, \ldots, n\}$))

4. Вычитание из конечного множества.

   $\forall_a$(конечное(a) $\rightarrow$ конечное($a \setminus b$))

5. Пересечение с конечным множеством.

   $\forall_{ab}$(конечное(a) $\rightarrow$ конечное($a \cap b$))

6. Объединение конечных множеств.

   $\forall_{ab}$(конечное(a) & конечное$(b) \to$ конечное$(a \cup b)$)

   $\forall_{ABn}(\forall_i(i \in \{1, \ldots, n\} \to$ конечное$(B(i)))$ & $A = \bigcup_{i=1}^{n} B(i) \to$ конечное$(A))$

   Во втором случае указатель "развертка(фикс(1)фикс(2 2))" определяет идентификацию объединения семейства с обычным объединением, а кванторной импликации в антецеденте - с конъюнкцией. Указатель "блокпроверок(1)" обрабатывает элементы этой конъюнкции проверочным оператором. Первый прием применяется в тех случаях, когда заголовком входного выражения служит символ "объединение"; второй - когда в посылках имеется равенство входного выражения объединению.

7. Подмножество конечного множества.

   $\forall_{ab}$(конечное(a) & $b \cup c \subseteq a \to$ конечное(b))

   Здесь второй антецедент идентифицируется непосредственно, причем указатель "единица(пусто x3)" допускает случай отсутствия c.

8. Конечность множества, для которого найдена числовая мощность.

   $\forall_{abcd}(\mathrm{card}(a \cup d) + b = c$ & b – число & c – число $\to$ конечное$(a))$

   Указатель "единица(0 x2)" допускает случай отсутствия слагаемого b.

9. Конечность прообраза.

   Если область определения отображения конечна, то прообраз элемента конечен:

   $\forall_{fABx}$(Отображение(f, A, B) & конечное(A) & $x \in B \to$ конечное(слой(f, x)))

10. Конечность пересечения промежутка с дискретным периодическим множеством.

    $\forall_{abpqr}(\neg(p = 0)$ & $\neg(r = 0) \to$ конечное$(\mathrm{set}_x(x \in [a, b]$ & $(px + q)/r$ – целое$)))$

11. Конечность прямого произведения.

    Рассмотрены случаи двух либо трех сомножителей:

    $\forall_{ab}$(конечное(a) & конечное$(b) \to$ конечное$(a \times b)$)

    $\forall_{abc}$(конечное(a) & конечное(b) & конечное$(c) \to$ конечное$(a \times b \times c)$)

12. Конечность множества, определяемого условным выражением.

    $\forall_{abc}$(конечное(a) & конечное$(b) \to$ конечное($(a$ при $c,$ иначе $b)$))

13. Конечность множества, заданного через описатель "класс".

    Выделены несколько специальных случаев, в которых усматривается конечность таких множеств:

    $\forall_{Aabcpq}(\neg(a = 0)$ & конечное$(\mathrm{set}_i(\neg(ai{+}b = c)$ & $A(q(i)))) \to$ конечное$(\mathrm{set}_i(A((p(i)$ при $ai + b = c,$ иначе $q(i))))))$

    $\forall_{Aabc}(\neg(a = 0) \to$ конечное$(\mathrm{set}_i(A(i)$ & $ai + b = c)))$

    В этом и предыдущем приемах используется одноэлементность множества корней линейного уравнения.

$\forall_{ABP}$(конечное(A) & конечное(B) $\rightarrow$ конечное(set$_{ij}(i \in A$ & $j \in B$ & $P(i,j))))$

$\forall_{ABP}$(конечное$(A(j))$ & конечное(B) $\rightarrow$ конечное(set$_{ij}(i \in A(j)$ & $j \in B$ & $P(i,j))))$

Антецедент $A(j)$ обрабатывается проверочным оператором, которому придается дополнительная посылка $j \in B$.

$\forall_{mnP}$(конечное(set$_x(x \in \{m,\ldots,n\}$ & $P(x))))$

$\forall_{mnp}$(конечное(set$_x(x$ – целое & $m \leq x$ & $x \leq n$ & $P(x))))$

$\forall_{An}(n$ – натуральное $\rightarrow$ конечное(set$_x$(двнабор(x,n) & $A(x))))$

14. Множество подмножеств конечного множества.

    $\forall_A$(конечное(A) $\rightarrow$ конечное(set$_B(B \subseteq A$ & $P(B))))$

15. Конечность множества, для которого известна верхняя числовая оценка мощности.

    $\forall_{ab}(\mathrm{card}(a) \leq b$ & b – число $\rightarrow$ конечное$(a))$

    Первый антецедент здесь идентифицируется непосредственно, второй - обрабатывается проверочным оператором. Указатель "вариант(фикс(1) меньше)" учитывает случай, когда неравенства для мощности строгое.

16. Конечность сомножителя конечного непустого прямого произведения.

    $\forall_{AB}(\mathrm{card}(A \times B) = a$ & $0 < a$ & a – число $\rightarrow$ конечное$(B))$

    Первый антецедент идентифицируется непосредственно. Аналогичный прием имеется для конечности сомножителя A.

17. Множество перестановок на конечном множестве.

    $\forall_A$(конечное(A) $\rightarrow$ конечное(set$_f$(перестановка(f,A) & $P(f))))$

18. Множество сочетаний.

    $\forall_{An}$(конечное(A) & n – натуральное $\rightarrow$ конечное(сочетания$(A,n)))$

    Напомним, что сочетания(A,n) обозначает множество всех подмножеств множества A, имеющих мощность n.

19. Конечность множества точек, заданных своими координатами.

    $\forall_{AK}$(конечное(A) $\rightarrow$ конечное(точки$(A,K)))$

    Напомним, что точки(A,K) обозначает множество точек, координаты которых в системе координат K принадлежат множеству A.

20. Конечность подмножества разрядов двоичного набора.

    $\forall_{ani}$(двнабор(a,n) $\rightarrow$ конечное(слой$(a,i)))$

21. Разное. Кроме перечисленных выше, проверочный оператор "усмконечное" имеет несколько приемов, связанных с графами, а также с неформальными понятиями, встречающимися в текстовых задачах.

Проверочный оператор "усмнеконечное"

Этот проверочный оператор усматривает бесконечность множества. Во-первых, он имеет приемы, предпринимающие попытку непосредственно вычислить мощность множества. Если результат - символ "счетное" либо "континуум", то множество бесконечно. Во-вторых, имеются приемы усмотрения бесконечности объединения либо разности множеств. В первом случае достаточно усмотрения бесконечности одного из объединяемых множеств, во втором - бесконечности "уменьшаемого" множества и конечности вычитаемого. Для усмотрени бесконечности прямого произведения проверяется бесконечность одного сомножителя и непустота другого. Наконец, введены приемы для усмотрения бесконечности конкретных числовых множеств (целые числа, четные, нечетные, простые).

Проверочные операторы "усмконечнпересек", "усмконечнпересечения"

Оператор "усмконечнпересек" проверяет, что пересечение множества A с любым множеством семейства множеств B конечно. Оператор "усмконечнпересечения" проверяет, что любые два различных элемента семейства множеств A имеют конечное пересечение. Первый оператор используется в приемах второго; второй - введен для приемов, вычисляющих длины кривых.

6.19 Упражнения

Ввести в задачник решателя перечисляемые ниже задачи по комбинаторике, переведя их на понятный ему логический язык. В тех случаях, когда на задачу выдается отказ, создать недостающие приемы и обеспечить ее решение.

1. Пять учеников следует распределить по трем параллельынм классам. Сколькими способами это можно сделать ?

2. Восемь авторов должны написать книгу из шестнадцати глав. Сколькими способами возможно распределение материала между авторами, если два человека напишут по три главы, четыре - по две и два - по одной главе книги ?

3. Семь яблок и три апельсина надо положить в два пакета так, чтобы в каждом пакете был хотя бы один апельсин и чтобы количество фруктов в них было одинаковым. Сколькими способами это можно сделать ?

4. Садовник должен в течение трех дней посадить 10 деревьев. Сколькими способами он может распределить по дням работу, если будет сажать не менее одного дерева в день ? Рассмотреть два случая, в одном из которых деревья не различаются, а в другом - различаются.

5. Сколькими способами можно выбрать 12 человек из 17, если данные двое человек из этих 17 не могут быть выбраны вместе ?

6. Сколькими способами можно выбрать из чисел от 1 до 100 три числа так, чтобы их сумма делилась на 3 ?

7. n предметов расположены в ряд. Сколькими способами можно выбрать из них три предмета так, чтобы не брать никаких двух соседних элементов ?

8. Сколькими способами 6 человек могут выбрать из 6 пар перчаток по правой и левой перчатке так, чтобы ни один не получил пары ?
9. Сколькими способами можно раздать 27 книг лицам A,B,C так, чтобы A и B вместе получили вдвое больше книг, чем C ?
10. Каких чисел от 1 до 10000000 будет больше: в записи которых встречается единица, или тех, в записи которых ее нет ?

6.19.1 Указания

1. Число способов распределения пяти учеников по трем классам равно числу отображений множества $\{1, \dots, 5\}$ (номера учеников) в множество $\{1, \dots, 3\}$ (номера классов). Вводим задачу на преобразование с целевой установкой "упростить выражение в области допустимых значений". Условием ее служит выражение $\mathrm{card}(\mathrm{set}_f(\text{Отображение}(f, \{1, \dots, 5\}, \{1, \dots, 3\})))$. Запускаем процесс решения и получаем ответ 243. При трассировке обнаруживаем, что сразу сработал прием, вычисляющий мощность класса отображений одного конечного множества в другое.

2. Задачу можно понимать двояко: если заранее известно, какие авторы сколько должны написать, то ответ будет одним; если это распределение допускается варьировать, то ответ другой. Ограничимся рассмотрением первого случая. Здесь требуется найти число таких отображений множества $\{1, \dots, 16\}$ в множество $\{1, \dots, 8\}$, что мощности прообразов элементов 1 и 2 равны 3; мощности прообразов элементов 3,4,5,6 равны 2; мощности прообразов элементов 7,8 равны 1. Это дает следующее условие задачи на преобразование:

   $\mathrm{card}(\mathrm{set}_f(\text{Отображение}(f, \{1, \dots, 16\}, \{1, \dots, 8\})\ \&\ \forall_i(i \in \{1, 2\} \to \mathrm{card}(\text{слой}(f, i)) = 3)\ \&\ \forall_i(i \in \{3, \dots, 6\} \to \mathrm{card}(\text{слой}(f, i))2)\ \&\ \forall_i(i \in \{7, 8\} \to \mathrm{card}(\text{слой}(f, i)) = 1)))$.

   Решатель выдает ответ 3632488000, что равно $(16!)/(2^6 \cdot 3^2)$. В принципе,при решении должна была бы применяться стандартная формула для числа отображений с заданными мощностями прообразов. Соответствующий прием легко найти в разделе "Непосредственное определение мощности". Однако, этот прием не имеет указателя "развертка", позволяющего идентифицировать список мощностей прообразов конкретных элементов. Он срабатывает лишь при условии, что мощности прообразов заданы некоторым общим выражением. При трассировке решения усматривается момент, когда условия на мощности прообразов перечисляются по отдельности для каждого элемента. Однако, ввиду отсутствия указанной версии приема, далее начинается трудоемкий процесс рекурсивного упрощения выражения, с рассмотрением каждого условия на мощность прообраза по отдельности. Рекомендуется самостоятельно добавить такую версию и обеспечить решение задачи с ее применением.

3. Формулировку задачи начнем с того, что обозначим множество апельсинов через A, множество яблок через B, и введем посылки $\mathrm{card}(A) = 3$, $\mathrm{card}(B) = 7$, непересек(A, B). Так как пакеты не различаются, то нужно найти множество неупорядоченных пар C, D подмножеств объединения $A \cup B$: в один из пакетов кладутся элементы C, в другой - элементы D. Дополнительные условия состоят в равенстве мощностей множеств C, D, в пересечении каждого из них с A,

а также в том, что эти множества дополняют друг друга до $A \cup B$. Отсюда получаем условие задачи:

$\mathrm{card}(\mathrm{set}_M(\exists_{CD}(M = \{C, D\} \ \& \ C \subseteq (A \cup B) \ \& \ \mathrm{card}(C) = \mathrm{card}(D) \ \& \ D = (A \cup B) \setminus C \ \& \ \neg(A \cap C = \emptyset) \ \& \ \neg(A \cap D = \emptyset))))$

Запускаем решатель и убеждаемся, что он задачу не решает. Выдаваемый ответ имеет вид:

$\mathrm{card}(\mathrm{set}_M(\exists_C(\neg(A \subseteq C) \ \& \ \neg(\text{непересек}(A, C)) \ \& \ M = \{C, (A \cup B) \setminus C\} \ \& \ \mathrm{card}((A \cup B) \setminus C) = \mathrm{card}(C) \ \& \ C - \mathrm{set}))).$

Так как не был выполнен переход от рассмотрения неупорядоченной пары M к рассмотрению упорядоченных пар, то естественно начать с создания приема, выполняющего этот переход. Выше был рассмотрен прием данного типа, у которого неупорядоченная пара относится к двум переменным связывающей приставки квантора существования. Здесь он неприменим. Будем создавать его обобщение на случай пар общего вида $\{p(y), q(y)\}$, где y - связывающая приставка квантора существования. Чтобы можно было перейти к неупорядоченным парам, следует, во-первых, убедиться, что $p(y)$ никогда не совпадает с $q(y)$, а также в том, что при перестановке $p(y)$ и $q(y)$ снова возникает допустимая пара, т.е. существует z, такое, что $p(z) = q(y)$ и $q(z) = p(y)$. Отсюда получаем теорему приема:

$\forall_{pqQ}(\neg(p(y) = q(y)) \ \& \ \exists_z(p(z) = q(y) \ \& \ q(z) = p(y) \ \& \ Q(z)) \to \mathrm{card}(\mathrm{set}_x(\exists_y(x = \{p(y), q(y)\} \ \& \ Q(y)))) = (\mathrm{card}(\mathrm{set}_{uv}(\exists_y(u = p(y) \ \& \ v = q(y) \ \& \ Q(y))))/2))$

Оба антецедента выделены указателями "следствие"; чтобы соответствующие вспомогательные задачи на доказательство получили дополнительные посылки $Q(y)$, введены указатели "занесениепосылки(1 значение(x41 x24))", "занесениепосылки(2 значение(x41 x24))". Прочие указатели приема - "отображение(x15 x16 x41)", "кортежпеременных(x24)", "внешнийквантор(фикс(0 1 1 2))", "список(фикс(0 1 1 2 2 1 2 1))". Компилируем прием и запускаем решение задачи. Созданного приема оказалось достаточно - решатель выдает ответ 105.

4. Ограничимся случаем, когда деревья не различаются. Тогда нужно найти число троек натуральных чисел x, y, z, составляющих в сумме 10: x деревьев будет посажено в первый день, y - во второй, и z - в третий. Условие задачи имеет следующий вид:

   $\mathrm{card}(\mathrm{set}_{xyz}(x - \text{натуральное} \ \& \ y - \text{натуральное} \ \& \ z - \text{натуральное} \ \& \ x + y + z = 10))$

   Решатель выдает ответ 36, используя прием для числа сочетаний с повторениями.

5. Обозначим через A множество из 17 человек; B - его подмножество из двух заданных человек. Тогда требуется определить число подмножеств C, не включающих B:

   $\mathrm{card}(\mathrm{set}_C(C - \mathrm{set} \ \& \ C \subseteq A \ \& \ \mathrm{card} C = 12 \ \& \ \neg(B \subseteq C)))$

   Вводим задачу и запускаем решение. Получаем ответ 3185.

6. Требуется найти число трехэлементных подмножеств A множества $\{1, \ldots, 100\}$, сумма которых делится на 3:

$$\mathrm{card}(\mathrm{set}_A(A - \mathrm{set}\ \&\ A \subseteq \{1, \ldots, 100\}\ \&\ \mathrm{card}A = 3\ \&\ 3| \sum_{i, i \in A} i))$$

На эту задачу решатель не находит ответа, и приходится создавать новые приемы. В учебнике по комбинаторике предлагается следующий план решения: тройка чисел представляется как объединение трех подмножеств, соответствующих различным вычетам по модулю 3, и анализируются возможные случаи для мощностей этих подмножеств. Попробуем реализовать его с помощью цепочки приемов. Прежде всего, нужно как-то объяснить переход к рассмотрению подмножеств, определяемых значениями вычета по модулю 3. Этот переход подсказывается тем, что в задаче имеется условие делимости конечной суммы на заданное натуральное число (3). Такую сумму можно заменить суммой произведений значений вычетов j на количества слагаемых, имеющих заданное значение вычета. Теорема приема такова:

$$\forall_{Atm}(m \mid \sum_{i, A(i)} t(i) \leftrightarrow m \mid \sum_{j=1}^{m-1}(j\mathrm{card}(\mathrm{set}_i(A(i)\ \&\ t(i)(\mathrm{mod}\ m) = j))))$$

Вводя фильтр "контекст(подчинено(теквхожд x2)символ(x2 мощность))", ограничиваем действие приема случаями, когда условие делимости расположено внутри выражения для мощности множества. Фильтр "натуральное(x13)" определяет идентификацию m с натуральной константой.

После применения данного приема возникают подвыражения $\mathrm{set}_i(i(\mathrm{mod}3) = 2\ \&\ i \in A)$, $\mathrm{set}_i(i(\mathrm{mod}3) = 1\ \&\ i \in A)$. Так как внешний описатель "класс" относится к перечисляемым множествам A, то удобно вынести из этих подвыражений множество A наружу, представив их как пересечения A с некоторыми не варьируемыми множествами. Для данной цели создаем следующий прием:

$\forall_{ABP}(A \subseteq B \to \mathrm{set}_i(i \in A\ \&\ P(i)) = A \cap \mathrm{set}_i(i \in B\ \&\ P(i)))$

Включение $A \subseteq B$ непосредственно встречается в контексте замены. Фильтр "контекст(подчинено(теквхожд x2)символ(x2 класс)входит(x26 связприставка(x2)) не(пересекаются(параметры(x27)связприставка(x2))))" указывает на то, что A есть варьируемая переменная внешнего описателя "класс", в то время как выражение B не имеет варьируемых переменных этого описателя.

После вынесения множества A наружу срабатывает уже имеющийся прием, представляющий это множество в виде $a \cup b$, где a - подмножество $\mathrm{set}_i(i(\mathrm{mod}3) - 2 = 0\ \&\ i \in \{1, \ldots, 100\}))$; b - подмножество дополнения последнего множества до $\{1, \ldots, 100\}$. Данный прием был приведен выше в подразделе "Параметрическое описание класса" раздела "Класс". Далее, после ряда несложных переходов, условие задачи приобретает следующий вид:

$\mathrm{card}(\mathrm{set}_{ab}(-3 + \mathrm{card}a + \mathrm{card}b = 0\ \&\ a - \mathrm{set}\ \&\ b - \mathrm{set}\ \&\ a \subseteq \mathrm{set}_i(-2 + i(\mathrm{mod}3) = 0\ \&\ i \in \{1, \ldots, 100\})\ \&\ b \subseteq \mathrm{set}_i(\neg(-2 + i(\mathrm{mod}3) = 0)\ \&\ i \in \{1, \ldots, 100\})\ \&\ 3|(2\mathrm{card}a + \mathrm{card}(b \cap \mathrm{set}_i(-1 + i(\mathrm{mod}3) = 0\ \&\ i \in \{1, \ldots, 100\})))))$

Снова срабатывает прием, подразбивающий b на два подмножества c и d,и реализуется цепочка упрощающих преобразований. В итоге имеем условие следующего вида:

$\mathrm{card}(\mathrm{set}_{cda}(-3+\mathrm{card}a+\mathrm{card}c+\mathrm{card}d=0\ \&\ a-\mathrm{set}\ \&\ c-\mathrm{set}\ \&\ d-\mathrm{set}\ \&\ a\subseteq \mathrm{set}_i(-2+i(\mathrm{mod}3)=0\ \&\ i\in\{1,\ldots,100\})\ \&\ c\subseteq \mathrm{set}_i(-1+i(\mathrm{mod}3)=0\ \&\ i\in\{1,\ldots,100\})\ \&\ d\subseteq \mathrm{set}_i(\neg(-1+i(\mathrm{mod}3)=0)\ \&\ \neg(-2+i(\mathrm{mod}3)=0)\ \&\ i\in\{1,\ldots,100\})\ \&\ 3|(2\mathrm{card}a+\mathrm{card}c)))$

Здесь возникает новая для решателя ситуация: нужно усмотреть, что числа, у которых значения вычета по модулю 3 отличны от 1 и 2, делятся на 3. Это усмотрение реализуем в два шага. Сначала обеспечиваем группировку отрицаний равенства вычета заданным значениям в виде отрицания принадлежности вычета. Инициализацию такой группировки выполняет следующий прием:

$\forall_{abckmn}(m=-a\ \&\ n=-c\ \&\ m\in\{0,\ldots,k-1\}\ \&\ n\in\{0,\ldots,k-1\}\to\neg(a+b(\mathrm{mod}k)=0)\ \&\ \neg(c+b(\mathrm{mod}k)=0)\leftrightarrow\neg(b(\mathrm{mod}k)\in\{m,n\}))$

В нашем случае нет необходимости создавать еще один прием, обеспечивающий слияние условий непринадлежности вычета и отрицания его равенства. Рекомендуется создать его самостоятельно. Указанный выше прием снабжаем фильтром "конец(контекст(подчинено(теквхожд x4)символ(x4 класс)пересекаются(x2 связприставка(x4))))", ограничиваясь пока лишь задачами по комбинаторике. После того, как отрицания условий равенства модуля преобразованы в условие непринадлежности модуля, оказывается необходим прием, усматривающий, что остается единственное возможное значение модуля. Вводим этот прием:

$\forall_{abcn}(l(b)=n-1\ \&\ \{0,\ldots,n-1\}\setminus\{;b\}=\{c\}\to\neg(a(\mathrm{mod}n)\in\{;b\})\leftrightarrow a(\mathrm{mod}n)=c)$

После срабатывания приема получаем такое условие задачи:

$\mathrm{card}(\mathrm{set}_{cda}(-3+\mathrm{card}a+\mathrm{card}c+\mathrm{card}d=0\ \&\ a-\mathrm{set}\ \&\ c-\mathrm{set}\ \&\ d-\mathrm{set}\ \&\ a\subseteq \mathrm{set}_i(-2+i(\mathrm{mod}3)=0\ \&\ i\in\{1,\ldots,100\})\ \&\ c\subseteq \mathrm{set}_i(-1+i(\mathrm{mod}3)=0\ \&\ i\in\{1,\ldots,100\})\ \&\ d\subseteq \mathrm{set}_i(i(\mathrm{mod}3)=0\ \&\ i\in\{1,\ldots,100\})\ \&\ 3|(2\mathrm{card}a+\mathrm{card}c)))$

Так как мощность каждого из множеств a, b, c не превосходит 3, причем фиксация этого значения позволяет декомпозировать условие под описателем на независимые подусловия для a, b, c, то нужен прием, разбирающий случаи для значения мощности. Отнесем его к какому-либо одному из параметров a, b, c. Чтобы избежать усложнения приема обращением к пакету, определяющему верхнюю оценку мощности множества, ограничимся ситуацией нашей задачи, где такая оценка извлекается из равенства суммы мощностей тройке. Получаем следующую теорему приема:

$\forall_{APmn}(\text{конечное}(A)\ \&\ 0\le m(y)\to\mathrm{card}(\mathrm{set}_{xy}(x-\mathrm{set}\ \&\ x\subseteq A\ \&\ \mathrm{card}x+m(y)-n=0\ \&\ P(\mathrm{card}x,y)))=\sum_{i=0}^{n}(C^{i}_{\mathrm{card}A}\mathrm{card}(\mathrm{set}_y(i+m(y)-n=0\ \&\ P(i,y)))))$

Указатель "развертка(фикс(0 2))" определяет преобразование конечной суммы в обычную. Чтобы выполнить идентификацию выражения $P(\mathrm{card}x, y)$, вводим указатель "новаргумент(x40 x23 фикс)". Необходим также указатель "кортежпеременных(x24)". Выражения для мощности под суммой обрабатываем вспомогательными задачами на преобразование. В нашей задаче это обеспечит рекурсивное срабатывание приема для разбора случаев по одному из оставшихся параметров a, b, c. Применение приема приводит к выдаче ответа: 53922.

7. Можно рассматривать задачу как нахождение числа троек целых чисел i, j, k, удовлетворяющих неравенствам $1 \leq i, i < j-1, j < k-1, k \leq n$. Тогда ее условие имеет следующий вид:

   card(set$_{ijk}$($1 \leq i$ & $i < j-1$ & $j < k-1$ & $k \leq n$ & $i-$целое & $j-$целое & $k-$целое))

   В посылки задачи заносим утверждение n – натуральное. Решатель выдает в качестве ответа сумму всех значений $(j-2)$ по множеству пар j, k, удовлетворяющих условиям $j \in \{3, \ldots, k-2\}, k$ – целое, $5 \leq k, k \leq n$. Это подсказывает необходимость создания приема, преобразующего двойную сумму в повторную. Теорема приема, например, может иметь следующий вид:

   $$\forall_{PQtm}(m(j) = \sum_{i,P(i,j)} t(i,j) \rightarrow \sum_{i,j,P(i,j)\ \&\ Q(j)} t(i,j) = \sum_{j,Q(j)} m(j))$$

   Указатель "перечень(x41 не(входит(x9 x41)))" определяет идентификацию утвер ждения $Q(j)$ со всеми конъюнктивными членами, не содержащими параметра i. Антецедент выполняет суммирование выражений $t(i,j)$ по i. Для этого используется вспомогательная задача на преобразование, посылки которой пополнены конъюнктивными членами утверждения $Q(j)$, т.е. вводится нормализатор "задача(4 упростить) посылки(значение(x41 x10))". Указатель "элемент(x9)" разрешает идентифицировать переменную i либо с первым, либо со вторым параметрами суммирования. Наконец, фильтр "не(входит(суммавсех значение(x13 x10)))" проверяет успех попытки суммирования по i. После создания приема решатель выдает ответ в следующем виде: "Если $5 \leq n$, то $(n-2)(n-3)(n-4)/6$, иначе 0".

8. Выбор левых перчаток представляет собой перестановку множества $\{1, \ldots, 6\}$ номеров пар: на i-м месте расположен номер пары, из которой берет перчатку i-й человек. Аналогичную перестановку имеем для правых перчаток. Соответственно, формулируем условие задачи:

   card(set$_{fg}$(перестановка$(f, \{1, \ldots, 6\})$ & перестановка$(g, \{1, \ldots, 6\})$ & $\forall_i(i \in \{1, \ldots, 6\} \rightarrow \neg(f(i) = g(i))$))))

   Запускаем процесс решения и обнаруживаем, что система "вязнет" в преобразованиях, не доходя до ответа. Входим в трассировку решения (клавиша "Щ"). Решатель сначала избавляется от квантора общности, заменяя его на шесть отрицаний равенства $\neg(f(i) = g(i))$. Затем начинаются попытки применить прием, представляющий искомую мощность как разность мощности множества с отброшенным отрицанием некоторого равенства и множества, где оно заменено на равенство. Таким образом удается дойти до задачи на вычисление мощности множества set$_{fg}$($f(6) = g(6)$ & перестановка$(f, \{1, \ldots, 6\})$ & перестановка$(g, \{1, \ldots, 6\})$). Эту мощность решатель не вычисляет. Так как значение перестановки f в точке 6 фиксировано, ее можно свести к перестановке меньшего множества. Соответствующий прием имеет вид:

   $\forall_{iAPt}$(card(set$_{fg}$(перестановка$(f, A(g))$ & $f(i) = t(g)$ & $P(f,g)$))) = card(set$_{fg}$(перестановка$(f, A(g) \setminus \{t(g)\})$ & P(Вставка$(f, i, t(g)), g$))))

   После срабатывания данного приема возникает выражение card(set$_{fg}$(перестановка$(f, \{1, \ldots, 6\} \setminus \{g(6)\})$ & перестановка$(g, \{1, \ldots, 6\})$))). Связь переменных

f, g пока не исчезла - она сохраняется в выражении для множества значений f. Однако, мощность этого множества не зависит от g. Чтобы выполнить декомпозицию, вводим такой прием:

$\forall_{APn}(\mathrm{card}A(g) = n \to \mathrm{card}(\mathrm{set}_{fg}(\text{перестановка}(f, A(g))\ \&\ P(g))) =$
$n! \cdot \mathrm{card}(\mathrm{set}_g(P(g))))$

Антецедент выделен указателем "идентификатор"; для вычисления мощности $A(g)$ используется вспомогательная задача. Фильтр "не(входит(x7 x14))" проверяет, что найденная мощность от g не зависит. Добавленных приемов уже достаточно, и после трудоемких вычислений решатель выдает ответ 190800. Заметим, что фактически вычисления реализуют примитивную версию принципа "включения и исключения", сводящуюся к многократному представлению множеств $\mathrm{set}_x(P(x)\ \&\ \neg(Q(x)))$ в виде разности $\mathrm{set}_x P(x)$ и $\mathrm{set}_x(P(x)\ \&\ Q(x))$. Реализация в решателе обычной версии данного принципа предоставляется читателю.

9. Распеределение книг между A, B, C определяется функцией, заданной на множестве $\{1, \ldots, 27\}$ и принимающей значения $\{1, 2, 3\}$. Поэтому условие задачи имеет следующий вид:

   $\mathrm{card}(\mathrm{set}_f(\text{Отображение}(f, \{1, \ldots, 27\}, \{1, \ldots, 3\})\ \&\ 2\mathrm{card}(\text{слой}(f, 3)) = \mathrm{card}(\text{прообраз}(f, \{1, 2\})))).$

   При запуске решателя выясняется, что происходит лишь тривиальная арифметическая стандартизация. Выражения для мощностей прообраза элемента 3 и прообраза множества $\{1, 2\}$ сохраняются неизмененными. Так как эти прообразы дополняют друг друга до области определения отображения, то можно ввести прием, выражающий один из них через другой:

   $\forall_{ABfc}(\text{Отображение}(f, C, A)\ \&\ B = A \setminus \{c\} \to \text{прообраз}(f, B) = C \setminus \text{слой}(f, c))$

   Инициализация приема происходит при усмотрении выражения "прообраз(f, B)". Указатель "контекст(позиция(x4 корень)вид(x4 слой(x6 x3)))" определяет следующим шагом идентификации усмотрение в текущем терме задачи подвыражения "слой(f, c)". Второй антецедент, выделенный указателем "идентификатор", проверяет, что множество B есть разность области значений A и одноэлементного множества $\{c\}$. После создания приема решатель определяет мощность прообраза элемента 3, и далее доводит задачу до ответа.

10. Множество чисел в указанном интервале, десятичная запись которых не содержит единицы, равномощно множеству отображений из $\{1, \ldots, 7\}$ в множество цифр $\{0, \ldots, 9\} \setminus \{1\}$, за исключением тождественно нулевого отображения. Поэтому условие задачи можно задать, например, следующим образом:

$$\mathrm{card}(\mathrm{set}_f(\text{Отображение}(f, \{1, \ldots, 7\}, \{0, \ldots, 9\} \setminus \{1\}))) - 1 < \frac{1}{2} \cdot 10000000$$

Тип задачи - "описать"; целевая установка - "проверить истинность утверждения". Ответ на задачу находится решателем сразу.

Глава 7

Приемы, связанные с числовыми множествами

В этом разделе рассматриваются приемы, относящиеся к числовым множествам. Большей частью, имеются в виду вещественные числа, хотя в отдельных случаях - также и комплексные. Перечислим основные понятия, встречающиеся в разделе. Начнем с констант, обозначающих конкретные числовые множества. "целыенеотрицательные" - множество целых неотрицательных чисел. Оно прорисовывается как **N**+, а вводится последовательным нажатием клавиш "Ц","Н". "натуральные" - множество натуральных чисел. Прорисовывается как **N**; вводится двукратным нажатием "N". "целые" - множество всех целых чисел. Прорисовывается в виде **Z**; вводится двукратным нажатием "Z". "рациональные" - множество всех рациональных чисел. Прорисовывается в виде **Q**; вводится двукратным нажатием "Q".

Числовой промежуток с концами a, b обозначается "промежуток(a b c d)", где c, d - индикаторы принадлежности промежутку его концов (c - для конца a, d - для b). Если конец включается в промежуток, то индикатор равен 1, иначе - равен 0. Концы a, b могут быть равны $-\infty$, ∞. В этих случаях соответствующий индикатор равен 0. Прорисовывается промежуток с принадлежащими ему концами в виде $[a, b]$; если конец не входит в промежуток, то квадратная скобка заменяется круглой. Если вместо конкретной константы (0 либо 1) роль индикатора играет некоторое выражение, то прорисовывается квадратная скобка. Выделим важный частный случай - совпадение промежутка со всей числовой прямой. Во внутреннем представлении имеем выражение "промежуток(минусбеск плюсбеск 0 0)", причем прорисовывается оно как **R**. Заметим, что такое выражение можно ввести двукратным нажатием "R". Предикат "числовойотрезок(x)" истинен, если x есть отрезок числовой прямой.

Конечный отрезок целых чисел, имеющий начало m и конец n, обозначается "номера(m n)". Прорисовывается он в виде $\{m, \ldots, n\}$. Для ввода с клавиатуры используется следующая последовательность действий: нажимается левая фигурная скобка; вводится выражение m; нажимаются клавиши "запятая", "Ctr-точка", "запятая"; вводится выражение n; нажимается правая фигурная скобка.

Множество членов арифметической прогрессии с первым членом a и знаменателем b обозначается "арифмпрогрессия(a b)".

Точные нижняя и верхняя грани числового множества A обозначаются "инф(A)" и "суп(A)". Прорисовываются они в виде $\inf A$, $\sup A$. Для ввода с клавиатуры в первом случае последовательно нажимаются "I,n"; во втором случае - "S,u". Утверждения "нижняягрань(m A)", "верхняягрань(m A)" означают, что число m, соответственно, не больше либо не меньше любого элемента числового множества A. В случае строгих неравенств используются утверждения "Нижняягрань(m A)", "Верхняягрань(m A)". Утверждения "наибольший(m A)", "наименьший(m A)" означают, что m - наибольший либо, соответственно, наименьший элемент числового множества A. Для ввода символов "нижняягрань". "Нижняягрань" последовательно нажимаются клавиши "н" ("Н") и "г". В случае верхних граней используются клавиши "в" ("В"), "г". Символы "наибольший", "наименьший" вводятся нажатием клавиш "Н","О" и "Н","Е". Утверждения "огрснизу(A)", "огрсверху(A)" означают, что числовое множество A ограничено, соответственно, снизу либо сверху. Для ввода с клавиатуры в первом случае последовательно нажимаются "г","н"; во втором случае - "г","в".

Выражение "внутренность(A)" обозначает множество внутренних точек множества A. Допустимы различные случаи: множество вещественных чисел; числовых наборов; точек плоскости либо пространства; комплексных чисел. В каждом случае работают свои приемы. Выражение "Внутренность(A)" обозначает расширенную внутренность числового множества A, к которой относятся также точки $-\infty$, ∞, если некоторые их окрестности содержатся в A. Выражение "граница(A)" обозначает множество граничных точек множества A. Выражение "замыкание(A)" обозначает замыкание множества A. Утверждение "замкнутое(A)" означает, что множество A замкнуто. Утверждение "Область(A)" означает, что A есть связное открытое множество. Утверждение "предельнточка(m A)" означает, что m есть предельная точка множества A. Утверждение "компсвязности(A B)" означает, что B есть компонента связности множества A. Во всех случаях допускаются множества перечисленных выше типов.

Утверждение "разделены(S)" означает, что S есть семейство подмножеств евклидова пространства, любые два из которых пересекаются по множеству меры ноль. Это понятие используется в решателе при вычислении двойных интегралов по объединению множеств. Утверждение "отделено(A S)", введенное как вспомогательное для усмотрения предыдущего утверждения, означает, что подмножество евклидова пространства A пересекается с любым подмножеством евклидова пространства, относящимся к семейству S, по множеству меры ноль.

Чтобы иметь возможность определять область двумерного евклидова пространства по ограничивающей ее кривой, введено выражение "областьграницы(A)". Здесь A - множество точек ограничивающей кривой.

Наконец, упомянем совсем редко используемое понятие "циклначало". Выражение "циклначало(a n)" обозначает первый в циклическом порядке, определенном на множестве $\{1, \ldots, n\}$, элемент группы a идущих подряд (допускается переход к 1 после n) значений. Предполагается, что a - собственный циклический подотрезок отрезка $\{1, \ldots, n\}$.

7.1 Приемы символа "промежуток"

Развертка принадлежности числовому промежутку

Обычно условие принадлежности элемента числовому промежутку переформулируется с помощью неравенств, причем это происходит уже на уровне 0. Для такой переформулировки предусмотрен ряд приемов, отличающихся друг от друга условиями срабатывания и группировкой членов неравенств. Начнем со следующего приема:

$$\forall_{abc}(b - \text{число} \ \& \ c - \text{число} \rightarrow a \in (b, c) \leftrightarrow a - \text{число} \ \& \ 0 < a - b \ \& \ 0 < c - a)$$

В скобочной записи выражение (a, b) имеет вид "промежуток(x1 x2 0 0)". Чтобы прием срабатывал и для промежутков других типов, введены указатели "альтернатива(фикс(0 1 2 3)1 фикс(0 2 2)меньшеилиравно)", "альтернатива(фикс(0 1 2 4)1 фикс(0 2 3)меньшеилиравно)". Они разрешают одновременную замену указателя типа конца промежутка (0 на 1) и соответствующего строгого неравенства на нестрогое. Фильтр "или(не(тип(описать))известно(x1) не(известно(x2))не(известно(x3)))" блокирует применение приема в тех случаях, когда решается задача на описание, причем утверждается принадлежность неизвестного выражения известному промежутку. Фильтры "или(не(тип(описать))не(цель(вычисление)))", "не(контекст(подчинено(теквхожд x4)символ(x4 отображение)входит(x1 связприставка(x4))операнд(x5 x4) символ(x5 интеграл)))" были введены в прием при проработке последующих разделов. Они отсекают случаи, где прием требует либо другой группировки членов неравенства, либо особого описания контекста срабатывания. В этих случаях работают другие приемы (см. ниже). Фильтр "условие" ограничивает действие приема лишь условиями задач.

К рассмотренному приему примыкают еще три, относящиеся к промежуткам с бесконечными концами:

$\forall_a(a \in \mathbf{R} \leftrightarrow a - \text{число})$

$\forall_{ab}(b - \text{число} \rightarrow a \in (b, \infty) \leftrightarrow a - \text{число} \ \& \ 0 < a - b)$

$\forall_{ab}(b - \text{число} \rightarrow a \in (-\infty, b) \leftrightarrow a - \text{число} \ \& \ 0 < b - a)$

Первый из приемов применяется без ограничений; фильтры двух последних аналогичны фильтрам приема, разобранного выше.

Далее идет группа приемов, у которых члены неравенств группируются в разных частях:

$\forall_{abc}(b - \text{число} \ \& \ c - \text{число} \rightarrow a \in (b, c) \leftrightarrow a - \text{число} \ \& \ b < a \ \& \ a < c)$

$\forall_{ab}(b - \text{число} \rightarrow a \in (b, \infty) \leftrightarrow a - \text{число} \ \& \ b < a)$

$\forall_{ab}(b - \text{число} \rightarrow a \in (-\infty, b) \leftrightarrow a - \text{число} \ \& a < b)$

В этих приемах предполагается, что преобразуется условие задачи на описание, причем выражение a содержит неизвестные, а выражение для промежутка неизвестных не содержит. Если условие принадлежности - единственное, содержащее неизвестную a, причем требуется найти пример, то прием блокируется. Для той же группировки членов неравенств созданы еще несколько приемов, относящихся к указанным выше особым случаям (синтез программ, вычисление интегралов, посылки задач на исследование).

Усмотрение серии промежутков

Если серия промежутков задана с помощью описателя "класс", то она переписывается в явном виде:

$$\forall_{fgh}(\text{set}_x(x - \text{число} \ \& \ \exists_n(f(n) \ \& \ g(n) < x \ \& \ x < h(n))) = \bigcup_{n, f(n)} (g(n), h(n))).$$

Указатель "внешнийквантор(фикс(0 1 2 2))" определяет идентификацию квантора существования без перехода к нему от отрицания квантора общности. Созданы еще три приема для альтернативных типов неравенств.

Коррекция концевых указателей для неограниченных промежутков

Чтобы устранять возможные мелкие неточности, когда бесконечный конец формально отнесен к промежутку, введены следующие два приема:

$\forall_{ab}(\text{промежуток}(-\infty, a, 1, b) = \text{промежуток}(-\infty, a, 0, b))$

$\forall_{ab}(\text{промежуток}(a, \infty, b, 1) = \text{промежуток}(a, \infty, b, 0))$

Промежуток с совпадающими концами

Если промежуток имеет совпадающие концы, причем хотя бы один из них к нему не отнесен, то промежуток пуст:

$\forall_{ab}(\text{промежуток}(a, a, 0, b) = \emptyset)$

$\forall_{ab}(\text{промежуток}(a, a, b, 0) = \emptyset)$

Правый конец меньше левого

В этом случае промежуток пуст:

$\forall_{abcd}(0 < a - b \to [a, b] = \emptyset)$

c, d - указатели типов концов, которые в формульной записи на экран не выводятся.

Объединение промежутков

Несколько приемов преобразуют объединение двух промежутков с общим концом к виду одного промежутка:

$\forall_{abc}([a, b) \cup (c, b) = ([a, b) \text{ при } 0 \leq c - a, \text{ иначе } (c, b)))$

$\forall_{abc}((b, a] \cup (b, c) = ((b, a] \text{ при } 0 \leq a - c, \text{ иначе } (b, c)))$

$\forall_{abcde}(0 < b - a \;\&\; 0 < c - b \to \text{промежуток}(a, b, d, 0) \cup \text{промежуток}(b, c, 1, e) = \text{промежуток}(a, c, d, e))$

$\forall_{abcde}(0 \leq b - a \;\&\; 0 \leq c - b \to \text{промежуток}(a, b, d, 1) \cup \text{промежуток}(b, c, 1, e) = \text{промежуток}(a, c, d, e))$

$\forall_{abcde}(0 < b - a \;\&\; 0 < c - b \to \text{промежуток}(a, b, d, 1) \cup \text{промежуток}(b, c, 0, e) = \text{промежуток}(a, c, d, e))$

В первых двух приемах индикаторы концов промежутков фиксированы так, как это прорисовано формульным редактором. Эти приемы имеют указатели, разрешающие концам b, c принадлежать промежуткам. Кроме перечисленных выше, в разделе имеется также прием, объединяющий два противоположно направленных открытых луча с общим концом:

$\forall_a((-\infty, a) \cup (a, \infty) = \mathbf{R} \setminus \{a\})$

Включение промежутков

Условие включения двух промежутков друг в друга выражается через неравенства для концов:

$\forall_{abcd}([a, b] \subseteq [c, d] \leftrightarrow c \leq a \;\&\; b \leq d \;\vee\; b - a < 0)$

$\forall_{abcd}([a, b] \subseteq [c, d) \leftrightarrow c \leq a \;\&\; b < d \;\vee\; b - a < 0)$

$\forall_{abcd}([a,b] \subseteq (c,d) \leftrightarrow c < a \ \& \ b < d \ \vee \ b - a < 0)$

$\forall_{abcd}([a,b] \subseteq (c,d] \leftrightarrow c < a \ \& \ b \leq d \ \vee \ b - a < 0)$

$\forall_{abcd}([a,b] \subseteq [c,d] \leftrightarrow c \leq a \ \& \ b \leq d \ \vee \ b - a \leq 0)$

$\forall_{abcd}([a,b) \subseteq (c,d] \leftrightarrow c < a \ \& \ b \leq d \ \vee \ b - a \leq 0)$

Приемы имеют указатели, учитывающие некоторые альтернативные возможности для индикаторов концов.

Равенство промежутков

Условие равенства двух невырожденных промежутков сводится к равенствам их концов и индикаторов концов:

$\forall_{abcdefgh}(0 < b - a \to \text{промежуток}(a,b,c,d) = \text{промежуток}(e,f,g,h) \leftrightarrow (a = e \ \& \ b = f \ \& \ c = g \ \& \ d = h)$

Два приема переформулируют условие пустоты промежутка в терминах неравенств:

$\forall_{ab}([a,b] = \emptyset \leftrightarrow 0 < a - b)$

$\forall_{ab}((a,b) = \emptyset \leftrightarrow 0 \leq a - b)$

В последнем случае добавлены указатели, разрешающие одному из концов относиться к промежутку.

Непересечение промежутков

Условие непересечения промежутков переформулируется в терминах неравенств для их концов:

$\forall_{abc}(\text{непересек}((a,b],[c,\infty)) \leftrightarrow 0 \leq a - b \ \vee \ 0 < c - b)$

$\forall_{abc}(\text{непересек}([a,b],[c,\infty)) \leftrightarrow 0 < a - b \ \vee \ 0 < c - b)$

Введены указатели, учитывающие случай невключения конца b в промежуток.

Добавление точки к концу промежутка

При добавлении точки к концу промежутка происходит изменение индикатора этого конца на 1:

$\forall_{ab}(0 < b - a \to \{a\} \cup (a,b) = [a,b))$

$\forall_{ab}(0 < b - a \to \{b\} \cup (a,b) = (a,b])$

Введен указатель, разрешающий альтернативный тип противоположного конца. В случае присоединения сразу двух точек создан следующий прием:
$\forall_{ab}(0 < b - a \to \{a,b\} \cup (a,b) = [a,b])$

Усмотрение числового множества

Если множества задано с помощью описателя "класс", причем имеется условие "число(x)" на варьируемый элемент множества, то условие включения такого множества в числовую ось заменяется на константу "истина":

$\forall_{fa}(a = \text{set}_x(x - \text{число} \ \& \ f(x)) \to a \subseteq \mathbf{R})$

Переход к меньшему промежутку

Условие принадлежности элемента промежутку и неравенство для этого элемента преобразуются в условие принадлежности элемента меньшему промежутку:

$\forall_{abcdef}(0 < d - b \ \& \ 0 < c - d \to a \in \text{промежуток}(b, c, e, f) \ \& \ a < d \leftrightarrow$
$a \in \text{промежуток}(b, d, e, 0))$

Свертка в условие принадлежности числовому промежутку

Переход от пары неравенств для числового параметра к условию принадлежности его числовому промежутку выполняется при решении задач на программирование для получения одной из стандартных вычислительных схем:

$\forall_{abc}(b - \text{число} \ \& \ c - \text{число} \to a \in [b, c] \leftrightarrow a - \text{число} \ \& \ b \le a \ \& \ a \le c)$

Замена выполняется справа налево; задача имеет тип "описать" и цель "программа". Предполагается наличие внешнего описателя "класс", связывающего переменную a.

7.2 Приемы символа "целыенеотрицательные"

Единственный прием этого раздела - расшифровка по определению условия принадлежности:

$\forall_n(n \in \mathbf{N}+ \leftrightarrow n - \text{целое} \ \& \ 0 \le n)$

7.3 Приемы символа "целые"

Кроме приемов символа "целые", в разделе имеется несколько приемов, относящихся к множествам целых чисел, заданным с помощью описателя "класс".

Расшифровка условия принадлежности

$\forall_z(z \in \mathbf{Z} \leftrightarrow z - \text{целое})$

Склейка двух примыкающих фрагментов множества целых чисел

$\forall_{mn}(n - m = 1 \to \text{set}_x(x - \text{целое} \ \& \ x \le m) \cup \text{set}_x(x - \text{целое} \ \& \ n \le x) = \text{set}_x(x - \text{целое}))$

Антецедент выделен указателем "идентификатор".

Вычитание из множества целых чисел множества натуральных чисел

$\forall_A(\mathbf{Z} \setminus (A \cup \mathbf{N}) = \text{set}_n(n - \text{целое} \ \& \ n \le 0) \setminus A)$

Указатель "единица(пусто x26)" разрешает вырожденный случай отсутствия множества A.

Отбрасывание конца целочисленного луча

$\forall_n(n - \text{целое} \to \text{set}_m(m - \text{целое} \ \& \ m \le n) \setminus \{n\} = \text{set}_m(m - \text{целое} \ \& \ m \le n - 1))$

Вычитание подмножества целых чисел, определенного с помощью описателя "класс"

$\forall_{AP}(\mathbf{Z} \setminus (A \cup \mathrm{set}_x(x - \text{целое } \& \ P(x))) = \mathrm{set}_x(x - \text{целое } \& \ \neg(P(x)) \setminus A))$

Множество A может отсутствовать.

Вычитание из подмножества целых чисел подмножества натуральных чисел

$\forall_{ABC}(\mathrm{set}_x(x-\text{целое } \& \ B(x)) \setminus (A \cup \mathrm{set}_y(y-\text{натуральное } \& \ C(y))) = \mathrm{set}_x(x-\text{целое } \& \ B(x) \ \& \ (x \leq 0 \ \vee \ \neg(C(x))) \setminus A))$

7.4 Приемы символа "натуральные"

Расшифровка условия принадлежности

$\forall_n(n \in \mathbf{N} \leftrightarrow n - \text{натуральное})$

Вычитание бесконечного луча

$\forall_n(n - \text{натуральное} \rightarrow \mathbf{N} \setminus \{n, \ldots, \infty\} = \{1, \ldots, n-1\})$

Вычитание множества простых чисел

$\mathbf{N} \setminus \mathrm{set}_x(\text{простое}(x)) = \mathrm{set}_x(x - \text{натуральное } \& \ \neg(\text{простое}(x)))$

Вычитание подмножества натуральных чисел, определенного с помощью описателя "класс"

$\forall_P(\mathbf{N} \setminus \mathrm{set}_x(x - \text{натуральное } \& \ P(x)) = \mathrm{set}_x(x - \text{натуральное } \& \ \neg(P(x))))$

7.5 Приемы символа "рациональные"

Пока был создан единственный прием - расшифровка условия принадлежности:

$\forall_a(a \in Q \leftrightarrow a - \mathrm{rational})$

7.6 Приемы символа "номера"

Условие пустоты конечного отрезка целочисленного ряда

$\forall_{mn}(m - \text{целое } \& \ n - \text{целое} \rightarrow \{m, \ldots, n\} = \emptyset \leftrightarrow n < m)$

$\forall_{mn}(0 < n - m \rightarrow \{n, \ldots, m\} = \emptyset)$

Развертка принадлежности конечному отрезку целых чисел

Как и в случае условия принадлежности числовому промежутку, группировка членов неравенства зависит от контекста. Если нет особых причин размещать ненулевые члены по обе стороны от знака неравенства, то они группируются справа:

$\forall_{abc}(a - \text{целое} \ \& \ b - \text{целое} \ \& \ c - \text{целое} \rightarrow a \in \{b, \dots, c\} \leftrightarrow 0 \leq a - b \ \& \ 0 \leq c - a)$

Если a - неизвестная задачи на описание, встречающаяся в прочих условиях, то при развертке условия принадлежности выражение a остается в одной части неравенства и не группируется с прочими ненулевыми членами. Это же имеет место в ряде других случаев (задачи на доказательство; описатель "отображение").

Условие принадлежности конечному отрезку целых чисел небольшой константной длины преобразуется не к виду неравенств, а к виду дизъюнкции равенств:

$\forall_{abcd}(a = b - c + 1 \ \& \ b - \text{целое} \ \& \ c - \text{целое} \rightarrow d \in \{c, \dots, b\} \leftrightarrow \exists_e(e \in \{0, \dots, a-1\} \ \& \ d = c + e))$

Первый антецедент вычисляет длину отрезка a. Правая его часть обрабатывается нормализаторами общей стандартизации. Фильтр "натуральное(x1)" проверяет, что найденная длина задается десятичной натуральной константой, а фильтр "меньше(числзначение(x1)15)" - что эта длина меньше 15. Указатель "или(фикс(0 2)фикс(0 2 2 1))" определяет развертку квантора существования в дизъюнкцию. Каждый член этой дизъюнкции соответствует конкретному значению параметра e. Перечисление значений выполняется согласно утверждению $e \in \{0, \dots, a-1\}$, а текущий дизъюнктивный член имеет вид $d = c + e$. Преобразование блокируется, если выполнено одно из следующих условий:

1. Утверждение входит в условие задачи на описание, причем $\{c, \dots, b\}$ не содержащит неизвестных;
2. Утверждение расположено внутри описателя "класс", имеющего несколько связанных переменных, и в том числе переменную d. При этом a больше 3;
3. Утверждение входит в кванторную импликацию, являющуюся посылкой задачи на исследование;
4. Утверждение расположено внутри выражения "мощность" и имеет связанные внешними кванторами и описателями переменные;
5. d - переменная, для которой текущий терм задачи имеет вхождение выражения $f(d)$.

Разность двух конечных отрезков целых чисел

Для вычитания двух конечных отрезков целых чисел, имеющих общий конец, созданы следующие приемы:

$\forall_{abc}(a - \text{целое} \ \& \ b - \text{целое} \ \& \ c - \text{целое} \ \& \ 0 \leq c - b \ \& \ 0 \leq b - a + 1 \rightarrow \{a, \dots, c\} \setminus \{a, \dots, b\} = \{b + 1, \dots, c\})$

$\forall abc(a - \text{целое} \ \& \ b - \text{целое} \ \& \ c - \text{целое} \ \& \ 0 \leq c - b + 1 \rightarrow \{a, \dots, c\} \setminus \{b, \dots, c\} = \{a, \dots, b - 1\})$

Для вычитания из конечного отрезка целых чисел конечного списка, содержащего начало этого отрезка, введен еще один прием:
$\forall_{abc}(0 \leq b - a \ \& \ a - \text{целое} \rightarrow \{a, \dots, b\} \setminus \{a; c\} = \{a + 1, \dots, b\} \setminus \{; c\})$

Явное перечисление конечного отрезка целых чисел

Если концы конечного отрезка целых чисел суть десятичные константы, то он преобразуется к виду конечного списка, перечисляющего все элементы отрезка. Теорема приема имеет следующий вид:

$\forall_{ija}(a = \{i, \dots, j\} \to \{i, \dots, j\} = a)$

Заголовок приема - "второйтерм". Фильтры "целое(x9)", "целое(x10)" определяют идентификацию параметров i, j с числовыми константами. Антецедент выделен указателем "идентификатор". Его правая часть обрабатывается нормализатором "нормномера", который, собственно, и реализует преобразование выражения "номера($i\ j$)" в конечный список. Фильтр "меньше(вычитание(числзначение(x10)числзначение(x9)) 9)" ограничивает длину результирующего списка. Кроме того, применение приема блокируется в следующих случаях:

1. Преобразуемое выражение входит в кванторную импликацию, являющуюся посылкой задачи на исследование;
2. Преобразуемое выражение не расположено внутри выражения "мощность(...)" и не расположено внутри теоретико-множественных операций "пересечение", "объединение", "разность", имеющих среди своих операндов вхождение символа "перечень";
3. Преобразуемое выражение $\{i, \dots, j\}$ не является операндом отношения $x \in \{i, \dots, j\}$, где x - переменная, для которой текущий терм задачи имеет подтерм вида $f(x)$;
4. Преобразуемое выражение не расположено внутри выражений с заголовками "перестановка", "Отображение".

Переход к подотрезку целых чисел при наличии дополнительного неравенства

$\forall_{mnkxa}(k - \text{целое}\ \&\ m - \text{целое}\ \&\ n - \text{целое}\ \&\ 0 \leq k - m\ \&\ 0 \leq n - k \to x \in \{m, \dots, n\} \setminus a\ \&\ x \leq k \leftrightarrow x \in \{m, \dots, k\} \setminus a)$

$\forall_{mnkxa}(k - \text{целое}\ \&\ m - \text{целое}\ \&\ n - \text{целое}\ \&\ 0 \leq n - k\ \&\ 0 \leq k - m \to x \in \{m, \dots, n\} \setminus a\ \&\ k \leq x \leftrightarrow x \in \{k, \dots, n\} \setminus a)$

$\forall_{ijmn}(i \in \{m, \dots, n\} \to j \in \{m, \dots, n\}\ \&\ i < j \leftrightarrow j \in \{i + 1, \dots, n\})$

Определение неизвестных границ отрезка целых чисел

Если целочисленный отрезок известен, то его неизвестные концы определяются с помощью операций $\inf, \sup$:

$\forall_{ija}(a = \{i, \dots, j\} \leftrightarrow i = \inf(a)\ \&\ j = \sup(a)\ \&\ a = [\inf(a), \sup(a)] \cap \mathbf{Z})$

Прием имеет фильтры "тип(описать)", "известно(x1)", "или(не(известно(x9))не(известно(x10)))".

Добавление к отрезку целых чисел одной точки

$\forall_{mnk}(m -$ целое & $n -$ целое & $0 \leq n - k$ & $m - k = 1 \rightarrow \{k\} \cup \{m, \dots, n\} = \{k, \dots, n\})$

$\forall_{mnk}(m -$ целое & $n -$ целое & $0 \leq k - m$ & $k - n = 1 \rightarrow \{k\} \cup \{m, \dots, n\} = \{m, \dots, k\})$

Равенства в антецедентах выделены указателем "идентификатор"; они проверяют, что добавляемая точка непосредственно примыкает к целочисленному отрезку.

Исключение крайней точки

$\forall_{imn}(\neg(i - m = 0) \rightarrow i \in \{m, \dots, n\} \leftrightarrow i \in \{m + 1, \dots, n\})$

$\forall_{mni}(i \in \{m, \dots, n\}$ & $\neg(i = n) \leftrightarrow i \in \{m, \dots, n - 1\})$

Пересечение с промежутком

$\forall_{mnk}(k -$ целое & $m -$ целое & $n -$ целое & $0 \leq k - m - 1$ & $0 \leq n - k + 1 \rightarrow (-\infty, k) \cap \{m, \dots, n\} = \{m, \dots, k\})$

Переход к подотрезку при наличии дополнительного условия принадлежности

Следующий прием упрощает условие на варьируемую переменную i, связанную внешним описателем "класс":

$\forall_{ijmnpA}(j \in \{p, \dots, n\} \rightarrow i \in \{m, \dots, n\} \setminus A$ & $j \in \{i+1, \dots, n\} \leftrightarrow i \in \{m, \dots, j-1\} \setminus A)$

Переменная j не связана в преобразуемом терме задачи кванторами и описателями. Допускается вырожденное значение $\emptyset$ переменной A.

Усмотрение принадлежности конечному отрезку целых чисел

Если в посылках задачи на исследование встречается выражение $\{m, \dots, n\}$ для отрезка целых чисел, причем из неравенств для величины i усматривается ее принадлежность данному отрезку, то выводится следствие $i \in \{m, \dots, n\}$:

$\forall_{mni}(i -$ целое & $m -$ целое & $n -$ целое & $0 \leq i - m$ & $0 \leq n - i \rightarrow i \in \{m, \dots, n\})$

Указатель "нормзнака(x9 минус плюс)" обеспечивает идентификацию i в тех случаях, когда оно имеет несколько слагаемых. Неравенства идентифицируются непосредственно; первые три антецедента обрабатываются проверочным оператором.

Условие включения двух конечных отрезков целых чисел

Условие включения непустого отрезка целых чисел в другой отрезок целых чисел преобразуется в пару неравенств:

$\forall_{mnpq}(0 \leq n - m$ & $m -$ целое & $n -$ целое & $p -$ целое & $q -$ целое $\rightarrow \{m, \dots, n\} \subseteq \{p, \dots, q\} \leftrightarrow p \leq m$ & $n \leq q)$

Объединение двух кванторных тождеств для смежных отрезков целых чисел

Если в посылках задачи на исследование встречаются две кванторных импликации, определяющих выражение $P(i)$, первая - на отрезке целых чисел $\{m, \ldots, n\}$, а вторая - на отрезке $\{p, \ldots, q\}$, примыкающем к первому, то они объединяются в одну кванторную импликацию:

$\forall_{abmnpqP}(p - n = 1 \rightarrow \forall_i(i \in \{m, \ldots, n\} \rightarrow P(i) = a(i))\ \&\ \forall_j(j \in \{p, \ldots, q\} \rightarrow P(j) = b(j)) \leftrightarrow \forall_i(iin\{m, \ldots, q\} \rightarrow P(i) = (a(i) \text{ при } i \leq n, \text{ иначе } b(i))))$

Прием применяется, если $P(i)$ содержит неизвестные, а выражения $a(i), b(j)$ - не содержат. Антецедент выделен указателем "идентификатор"; он проверяет, что второй отрезок является непосредственным продолжением первого.

Развертка квантора общности по конкретному отрезку целых чисел

Если антецедентом кванторной импликации служит условие принадлежности варьируемой переменной конечному отрезку целых чисел, концы которого определены числовыми константами, то импликацию можно преобразовать в конъюнкцию утверждений, получаемых при различной фиксации значения этой переменной:

$\forall_{mnkP}(k = n - m + 1 \rightarrow \forall_i(i \in \{m, \ldots, n\} \rightarrow P(i)) \leftrightarrow \forall_j(j \in \{1, \ldots, k\} \rightarrow P(m + j - 1)))$

Фильтры "целое(x13)", "целое(x14)" определяют идентификацию параметров m, n с целочисленными константами; фильтр "меньше(x11 9)" - проверяет, что вычисленная в первом антецеденте длина k целочисленного промежутка не превосходит 8. Пока прием оказался полезен лишь в теории вероятностей, поэтому введены фильтры, ограничивающие его применение случаями, когда кванторная импликация задает значение вероятности. Указатель "развертка(фикс(0 2))" определяет построение заменяющего терма в виде конъюнкции утверждений, соответствующих различным значениям j.

Принадлежность суммы конечному отрезку целых чисел

Условие принадлежности суммы либо разности, операндом которых служит связанная переменная i внешнего описателя, преобразуется в условие принадлежности для i:

$\forall_{kmni}(k - \text{целое} \rightarrow k - i \in \{m, \ldots, n\} \leftrightarrow i \in \{k - n, \ldots, k - m\})$

Выделение подмножества конечного отрезка

Если описатель задает подмножество элементов целочисленного отрезка, выделенное дополнительным условием, то это подмножество переформулируется в виде конечного списка:

$\forall_{abnP}(n = b - a\ \&\ a - \text{целое}\ \&\ b - \text{целое} \rightarrow \mathrm{set}_x(x - \text{целое}\ \&\ a \leq x\ \&\ x \leq b\ \&\ P(x)) = \bigcup_{i=0}^{n}(\{a + i\} \text{ при } P(a + i), \text{ иначе } \emptyset))$

Фильтр "натуральное(x14)" проверяет, что разность между концами отрезка есть натуральная константа. Указатель "развертка(фикс(0 2))" определяет формирование конечного объединения в виде обычного объединения $n + 1$ членов. Нормализаторы обеспечивают упрощение этих членов, а также упрощение всего объединения. Если удалось устранить условные выражения, то результат имеет вид конечного списка.

Нормализатор "нормномера"

В нормализаторе имеются три приема, обеспечивающие преобразование не более чем 8-элементного целочисленного промежутка к виду конечного списка, а также усмотрение пустого промежутка.

7.7 Экстремальные элементы и грани

7.7.1 Приемы символа "нижняягрань"

При доказательстве утверждений вида "нижняягрань(a M)", означающих, что a не превосходит произвольного элемента числового множества M, используется нормализатор таких утверждений "нормнижняягрань". В него вошли следующие приемы (замена везде выполняется слева направо):

1. Нижняя грань образа объединения.

   $\forall_{abcf}(\text{нижняягрань}(a, \text{образ}(f, b \cup c)) \leftrightarrow \text{нижняягрань}(a, \text{образ}(f, b))\ \&\ \text{нижняягрань}(a, \text{образ}(f, c)))$

   Утверждения "нижняягрань(...)" в заменяющей части обрабатываются тем же самым нормализатором.

2. Нижняя грань для образа промежутка. Если функция имеет на подмножестве промежутка производную, то достаточно проверять условие нижней грани в концах промежутка и особых точках (производная не определена либо равна 0):

   $\forall_{abcfg}(\text{Производная}(f, g) \rightarrow \text{нижняягрань}(a, \text{образ}(f, [b, c])) \leftrightarrow a \leq f(b)\ \&\ a \leq f(c)\ \&\ \text{нижняягрань}(a, \text{образ}(f, [b, c] \backslash \text{Dom}(g)))\ \&\ \text{нижняягрань}(a, \text{образ}(f, \text{roots}(g, [b, c]))))$

   $\forall_{abcfg}(\text{Производная}(f, g) \rightarrow \text{нижняягрань}(a, \text{образ}(f, (b, c))) \leftrightarrow a \leq \underline{\lim}_{x \to b+0} f(x)\ \&\ a \leq \underline{\lim}_{x \to c-0} f(x)\ \&\ \text{нижняягрань}(a, \text{образ}(f, (b, c) \backslash \text{Dom}(g)))\ \&\ \text{нижняягрань}(a, \text{образ}(f, \text{roots}(g, (b, c)))))$

   Аналогичные приемы введены для случаев, когда один конец включается в промежуток, а другой - не включается. Утверждения "нижняягрань(...)" в заменяющей части рекурсивным образом обрабатываются тем же самым нормализатором.

3. Группировки для образов множеств корней производной и ее особых точек.

   $\forall_{abcfg}(\text{Производная}(f, g) \rightarrow \text{нижняягрань}(a, \text{образ}(f, b \setminus \text{Dom}(g)))\ \&\ \text{нижняягрань}(a, \text{образ}(f, c \setminus \text{Dom}(g))) \leftrightarrow \text{нижняягрань}(a, \text{образ}(f, (b \cup c) \setminus \text{Dom}(g))))$

   $\forall_{abcfg}(\text{Производная}(f, g) \rightarrow \text{нижняягрань}(a, \text{образ}(f, \text{roots}(g, b)))\ \&\ \text{нижняягрань}(a, \text{образ}(f, \text{roots}(g, c))) \leftrightarrow \text{нижняягрань}(a, \text{образ}(f, \text{roots}(g, b \cup c))))$

4. Использование посылки, явно задающей прообраз. Если рассматривается образ множества b, для которого в посылках имеется ячное выражение c - через описатель "класс", либо через конечный список, то в утверждении "нижняягрань(...)" выражение b заменяется на c:

   $\forall_{abcfd}(b = c \rightarrow d \,\&\, \text{нижняягрань}(a, \text{образ}(f, b)) \leftrightarrow d \,\&\, \text{нижняягрань}(a, \text{образ}(f, c)))$

   Условия на b, c определяются фильтрами "заголовок(х3 перечень класс пусто)", "не(заголовок(х2 перечень класс пусто))".

5. Нижняя грань для пустого множества.

   $\forall_{ab}(b \;\&\; \text{нижняягрань}(a, \emptyset) \leftrightarrow b)$

6. Нижняя грань для перечня. Условие нижней грани для конечного списка заменяется на систему неравенств для его элементов:

   $\forall_{abc}(\text{нижняягрань}(a, \{b; c\}) \leftrightarrow a \leq b \;\&\; \text{нижняягрань}(a, \{; c\}))$

   Утверждение "нижняягрань(...)" в заменяющей части само обрабатывается нормализатором "нормнижняягрань", так что прием за один шаг переходит к системе неравенств.

7. Нижняя грань для серии. Условие нижней грани для множества, определяемого с помощью целочисленной параметризации, переформулируется в виде кванторной импликации:

   $\forall_{fga}(\text{нижняягрань}(a, \text{set}_b(\exists_n(b = f(n) \;\&\; n - \text{целое} \;\&\; g(n)))) \leftrightarrow \forall_n(n - \text{целое} \;\&\; g(n) \rightarrow a \leq f(n)))$

8. Склейка односторонних пределов.

   $\forall_{abf}(a \leq \underline{\lim}_{x \rightarrow b-0} f(x) \;\&\; a \leq \underline{\lim}_{x \rightarrow b+0} f(x) \leftrightarrow a \leq \underline{\lim}_{x \rightarrow b} f(x))$

9. Устранение одностороннего предела для символа бесконечности.

   $\forall_{apf}(p \;\&\; a \leq \underline{\lim}_{x \rightarrow -\infty+0} f(x) \leftrightarrow p \;\&\; a \leq \underline{\lim}_{x \rightarrow -\infty} f(x))$

   $\forall_{apf}(p \;\&\; a \leq \underline{\lim}_{x \rightarrow \infty-0} f(x) \leftrightarrow p \;\&\; a \leq \underline{\lim}_{x \rightarrow \infty} f(x))$

Кроме нормализатора "нормнижняягрань", в разделе имеется несколько приемов сканирования задач. Прежде всего, приведем прием, усматривающий в кванторной импликации условие нижней грани:

$\forall_{ae}(\forall_c(c \in a \rightarrow e \leq c) \leftrightarrow \text{нижняягрань}(e, a))$

Перед заменой проверяется отсутствие комментариев и целей, делающих предпочтительным сохранение кванторов. Для обратного преобразования кванторной расшифровки создан ряд приемов, основанных на следующей теореме:

$\forall_{ab}(\text{нижняягрань}(b, a) \leftrightarrow b - \text{число} \;\&\; \forall_c(c \in a \rightarrow b \leq c))$

Кванторная расшифровка выполняется в условиях задач на доказательство, а иногда - также в условиях задач на описание. Для случая множества элементов последовательности предусмотрен отдельный прием:

$\forall_{af}$(последовательность$(f, \mathbf{R}) \to$ нижняягрань$(a, \mathrm{Val}(f)) \leftrightarrow a$ – число & $\forall_n(n$ – натуральное $\to a \leq f(n)))$

Этот прием применяется к условию задачи на описание, имеющей цель "пример". Если в множестве, для которого указана нижняя грань b, выделены какие-либо элементы c, то выводятся следствия - неравенства для b и c:

$\forall_{abc}(a \subseteq \mathbf{R}$ & $c \in a$ & нижняягрань$(b, a) \to b \leq c)$

Наконец, упомянем прием вывода, усматривающий в нижней грани, принадлежащей множеству, его наименьший элемент:

$\forall_{ab}(b \subseteq \mathbf{R}$ & $a \in b$ & нижняягрань$(a, b) \to$ наименьший$(a, b))$

7.7.2 Приемы символа "Нижняягрань"

Для случая строгой нижней грани рассматривается символ "Нижняягрань". С ним связан нормализатор "нормНижняягрань", аналогичный описанному выше нормализатору "нормнижняягрань".

7.7.3 Приемы символа "верхняягрань"

Для символа "верхняягрань" нормализатор создан не был. Имеются приемы кванторной свертки, развертки и вывода, аналогичные случаю нижней грани. Кроме них, созданы еще два приема вывода, применяемые в задачах на исследование, имеющих цель "противоречие". Первый усматривает условие верхней грани в кванторной импликации:

$\forall_{Pa}(\forall_x(P(x) \to x \leq a) \to$ верхняягрань$(a, \mathrm{set}_x(P(x))))$

Второй предпринимает попытку усмотреть противоречивость условия верхней грани, подобрав в множестве элемент, больший ее:

$\forall_{ab}(b \subseteq \mathbf{R}$ & $\neg(b = \emptyset)$ & верхняягрань(a, b) & $\exists_x(x \in b$ & $a < x) \to$ ложь$)$

Четвертый антецедент обрабатывается вспомогательной задачей на доказательство. Указатель "лимит(2000000)" ограничивает трудоемкость этой попытки, делая ее сравнительно быстрой. Указатель "попытка(суп противоречие x2)" блокирует повторные попытки применения приема к множеству b.

7.7.4 Приемы символа "наибольший"

1. Расшифровка по определению.

   $\forall_{ab}$(наибольший$(a, b) \leftrightarrow a \in b$ & верхняягрань$(a, b))$

   Для этой теоремы созданы два приема. Первый применяется к условию задачи на описание, имеющей цель "развертка". Он срабатывает на уровне 1. Другой применяется к условию задачи на доказательство и срабатывает на уровне 6 либо 7. В случае посылок расшифровка применяется только к отрицанию утверждения "наибольший(...)"; уровень срабатывания здесь равен 5.

2. Свертка по определению.

   Теорема приема - та же, что в предыдущем пункте, но замена выполняется справа налево. Преобразование применяется к условию задачи на описание, имеющей цель "редуцирование".

3. Принадлежность наибольшего элемента множеству. Прием выполняет вывод следствий в посылках:

   $\forall_{ab}(\text{наибольший}(a,b) \rightarrow a \in b)$

4. Если верхняя грань принадлежит множеству, то она является наибольшим элементом. Прием выполняет вывод следствий в посылках:

   $\forall_{ab}(a \in b \ \& \ \text{верхняягрань}(a,b) \rightarrow \text{наибольший}(a,b))$

   Первый антецедент выделен указателем "блокпроверок".

5. Наибольший элемент является верхней гранью. Прием выполняет вывод следствий:

   $\forall_{ab}(\text{наибольший}(a,b) \rightarrow \text{верхняягрань}(a,b))$

7.7.5 Приемы символа "наименьший"

Приемы аналогичны приемам предыдущего подраздела.

7.7.6 Приемы символа "инф"

Символы "инф" и "суп" используются в решателе достаточно часто. Даже если множество имеет наименьший либо наибольший элементы, для его обозначения посредством выражения используются именно они.

1. Расшифровка по определению. Прием дает кванторную расшифровку равенства, в одной части которого содержится линейная комбинация с "инф" -ом:

   $\forall_{acde}(\neg(c=0) \rightarrow c\inf(a)+d=e \leftrightarrow \text{нижняягрань}((e-d)/c,a) \ \& \ \forall_x((e-d)/c < x \ \& \ x - \text{число} \rightarrow \exists_y(y \in a \ \& \ y < x)))$

   Эта замена выполняется в условии задачи на доказательство, и возникающая после расшифровки кванторная импликация легко устраняется. Если идентификация неоднозначна, выбирается самое длинное выражение $\inf(a)$ либо $\sup(a)$.

2. Кванторная свертка.

   $\forall_{ac}(\forall_d(d \in a \rightarrow c < d) \leftrightarrow c \leq \inf(a) \ \& \ \neg(c \in a))$

   Предполагается, что в контексте замены уже встречается выражение $\inf(a)$.

3. Объединение множеств.

   $\forall_{ab}(a \subseteq \mathbf{R} \ \& \ b \subseteq \mathbf{R} \ \& \ \neg(a=\emptyset) \ \& \ \neg(b=\emptyset) \rightarrow \inf(a \cup b) = \min(\inf(a),\inf(b)))$

   Замена выполняется слева направо и применяется к подвыражению условия задачи на преобразование.

4. Объединение семейства множеств.

   $$\forall_{ABa}(a(i)=\inf(B(i)) \ \& \ \neg(B(i)=\emptyset) \rightarrow \inf \bigcup_{i,A(i)} B(i) = \inf(\text{set}_x(\exists_i(A(i) \ \& \ x=a(i)))))$$

Первый антецедент выделен указателем "идентификатор". Он присваивает функциональной переменной $a(i)$ результат упрощения выражения $\inf(B(i))$. Проверяется, что этот результат не содержит символов "суп", "инф".

5. Числовые промежутки.

   $\forall_{abcd}(0 < a - b \rightarrow \inf(\text{промежуток}(b, a, c, d)) = b)$

6. Конечные множества.

   Для одноэлементного множества точная нижняя грань находится непосредственно:

   $\forall_a(\inf(\{a\}) = a)$

   В случае более чем одноэлементного конечного списка используется переход к меньшему списку:

   $\forall_{ab}(\inf(\{a; b\}) = \min(a, \inf(\{; b\})))$

   Так как фрагменты заменяющего терма обрабатываются нормализаторами "норминф", "нормминимум", то фактически точная нижняя грань определяется здесь за один шаг. В случае конечного отрезка целых чисел применяется следующий прием:

   $\forall_{mn}(0 \leq n - m \rightarrow \inf(\{m, \ldots, n\}) = m)$

7. Числовая последовательность.

   (a) Бесконечный нижний предел. Если нижний предел числовой последовательности равен $-\infty$, то точная нижняя грань множества ее членов тоже равна $-\infty$:

   $\forall_f(\underline{\lim}_{n\rightarrow\infty}\{f(n)\} = -\infty \rightarrow \inf(\text{set}_x(\exists_n(n - \text{натуральное} \ \& \ x = f(n)))) = -\infty)$

   Напомним, что фигурные скобки после предела означают рассмотрение натурального аргумента.

   (b) Анализ локальных минимумов и нижнего предела. Определение точной нижней грани множества членов числовой последовательности сводится к определению точной нижней грани множества ее значений в точках локальных минимумов, пополненного нижним пределом:

   $\forall_{fpA}((n - \text{натуральное} \ \& \ (n = 1 \vee 2 \leq n \ \& \ 0 \leq f(n-1) - f(n)) \ \& \ 0 \leq f(n+1) - f(n)) = A(n) \ \& \ p = \underline{\lim} n \rightarrow \infty\{f(n)\} \rightarrow \inf(\text{set}_x(\exists_n(n - \text{натуральное} \ \& \ x = f(n)))) = \inf(\text{set}_x(\exists_n(A(n) \ \& \ x = f(n))) \cup \{p\}))$

   Оба антецедента выделены указателем "идентификатор". Первый из них определяет условие $A(n)$ на точку локального минимума. Это условие явно разрешается вспомогательной задачей на описание относительно n. Второй антецедент вычисляет нижний предел p.

   (c) Разбиение на подпоследовательности, соответствующие константному значению тригонометрического подвыражения. Если общий член последовательности $f(n)$ задается с использованием тригонометрического выражения от аргумента $qn\pi/p$, где p, q - натуральные, то для определения точной нижней грани удобно перейти к рассмотрению подпоследовательностей, у которых данное тригонометрическое выражение имеет константное значение. В случаях синуса либо косинуса рассматриваются $2p$ подпоследовательностей; в случаях тангенса либо котангенса - p подпоследовательностей:

$\forall_{fp}(\inf(\mathrm{set}_x(\exists_n(n-$натуральное & $x=f(n)))) = \inf(\mathrm{set}_m(\exists_k(k \in \{0,\ldots,2p-1\}$ & $m = \inf(\mathrm{set}_x(\exists_n(n-$ натуральное & $x = f(2pn-k))))))))$

$\forall_{fp}(\inf(\mathrm{set}_x(\exists_n(n-$ натуральное & $x=f(n)))) = \inf(\mathrm{set}_m(\exists_k(k \in \{0,\ldots,p-1\}$ & $m = \inf(\mathrm{set}_x(\exists_n(n-$ натуральное & $x = f(pn-k))))))))$

В первом случае прием имеет указатель "контекст(тригаргумент(значение(x6 x14)x4)операнд(x5 x4)символ(x5 синус косинус)вид(x4 дробь(умножение(x16 x14 пи)x15))единица(1 x15 x16))". Во втором - такой же указатель, где вместо синуса и косинуса стоят тангенс и котангенс. Таким образом идентифицируются p и q. Фильтры "натуральное(x15)", "натуральное(x16)" уточняют, что p, q идентифицированы с натуральными константами. Указатель "или(фикс(0 2 1 2)фикс(0 2 1 2 2 1))" определяет развертку квантора существования в дизъюнкцию равенств переменной m найденным с помощью вспомогательной задачи значениям точной нижней грани для подпоследовательностей.

(d) Рассмотрение подпоследовательностей, на которых показатель степени с отрицательным основанием имеет постоянную четность. Прием аналогичен предыдущему, однако вместо тригонометрического подвыражения имеется степень с отрицательным основанием. Рассматриваются подпоследовательности, для которых показатель данной степени имеет постоянную четность. Это позволяет вынести минус из-под степени и упростить задачу поиска точной нижней грани. Созданы два приема. Первый гарантирует определение четности показателя степени:

$\forall_{fc}(\inf(\mathrm{set}_x(\exists_n(n-$ натуральное & $x=f(n)))) = \inf(\mathrm{set}_m(\exists_k(k \in \{0,\ldots,c-1\}$ & $m = \inf(\mathrm{set}_x(\exists_n(n-$ натуральное & $x = f(cn-k))))))))$

Указатель "контекст(позиция(x4 фикс(0 1 1 2 2 2 2))вид(x4 степень(минус(x5)плюс(x1 дробь(умножение(x14 x2)x3))))единица(1 x3)единица(0 x1 x2))" идентифицирует внутри $f(n)$ вхождение подвыражения $(-e)^{a+bn/c}$. Фильтры "натуральное(x3)", "целое(x1)", "целое(x2)" обеспечивают идентификацию a, b, c с целочисленными константами; фильтр "меньше(числзначение(x3)5)" требует, чтобы c не превосходило 4. Далее квантор существования разворачивается в дизъюнкцию аналогично тому, как это происходило в предыдущем пункте. Второй прием ориентирован на ситуации, в которых априори нет гарантий исключения минуса из основания степени. Здесь отдельно рассматриваются четные и нечетные значения индекса n, и замена предпринимается лишь при условии, что в каждом случае удалось вычислить точную нижнюю грань:

$\forall_{fpq}(p = \inf(\mathrm{set}_x(\exists_n(n-$ натуральное & $x = f(2n))))$ & $q = \inf(\mathrm{set}_x(\exists_n(n-$ натуральное & $x = f(2n-1)))) \rightarrow \inf(\mathrm{set}_x(\exists_n(n-$ натуральное & $x = f(n)))) = \min(p,q))$

Указатель "контекст(позиция(x4 значение(x6 x23))вид(x4 степень(минус(x5)умножение(плюс(x23 x3)x1)))единица(1 x1)единица(0 x3))" определяет идентификацию подвыражения $(-e)^{(x+c)a}$, причем никаких предположений о виде множителя a не делается.

(e) Учет монотонности. В случае неубывающей последовательности точная нижняя грань равна ее первому элементу:

$\forall_{an}(n$ – целое & неубывает$(\lambda_i(a(i), i$ – число$), \mathrm{set}_i(i$ – целое & $n \leq i)) \rightarrow \inf(\mathrm{set}_x(\exists_i(i$ – целое & $n \leq i$ & $x = a(i)))) = a(n))$

8. Вывод следствий. Если в посылках имеется равенство, задающее точную нижнюю грань множества, то для каждого выделенного в этом множестве элемента добавляется соответствующее неравенство:

   $\forall_{abc}(\inf(a) = b$ & $c \in a \rightarrow b \leq c)$

   Если в задаче упоминается точная нижняя грань множества, причем имеется посылка, выражающая свойство ограниченности этого множества снизу, то регистрируется посылка, что точная нижняя грань является нижней гранью:

   $\forall_a(a \subseteq \mathbf{R}$ & огрснизу$(a) \rightarrow$ нижняягрань$(\inf(a), a))$

   Если, кроме того, посылка выделяет какую-то нижнюю грань, то выводится неравенство, что последняя не превосходит точной нижней грани:

   $\forall_{ae}(a \subseteq \mathbf{R}$ & нижняягрань(e, a) & огрснизу$(a) \rightarrow 0 \leq \inf(a) - e)$

   Наконец, если упоминается точная нижняя грань и выделен элемент множества, то выводится неравенство для их разности:

   $\forall_{ab}(a \subseteq \mathbf{R}$ & огрснизу(a) & $b \in a \rightarrow 0 \leq b - \inf(a))$

9. Расшифровка равенства точной нижней грани множества значений численной характеристики $f(y)$ объектов характеристике конкретного объекта $f(a)$. Данное равенство эквивалентно условию, выражающему, что для всех допустимых y величина $f(a)$ не превосходит $f(y)$:

   $\forall_{fga}(g(a) \rightarrow \inf(\mathrm{set}_x(\exists_y(x = f(y)$ & $g(y)))) = f(a) \leftrightarrow \forall_y(g(y) \rightarrow f(a) \leq f(y)))$

10. Двухэлементный набор, для которого известны минимум и максимум. В этой ситуации симметричные операции над элементами набора переформулируются в терминах его меньшего и большего элементов:

    $\forall_{abc}(l(a) = 2$ & $\inf\{;a\} = b$ & $\sup\{;a\} = c \rightarrow a(1)a(2) = bc)$

    $\forall_{abc}(l(a) = 2$ & $\inf\{;a\} = b$ & $\sup\{;a\} = c \rightarrow a(1) + a(2) = b + c)$

    $\forall_{abc}(l(a) = 2$ & $\inf\{;a\} = b$ & $\sup\{;a\} = c \rightarrow \{a(1), a(2)\} = \{b, c\})$

    Предполагается, что a - переменная, не входящая в b, c.

11. Достижение функцией наименьшего значения в заданной точке. Условие переформулируется в терминах неравенств:

    $\forall_{fa}(f(a) = \inf(\mathrm{Val}(f)) \leftrightarrow \forall_x(x \in \mathrm{Dom}(f) \rightarrow f(a) \leq f(x)))$

12. Исключение несущественной неизвестной. Если рассматривается числовая последовательность f, то для конечного множества ее членов всегда можно выбрать такое натуральное n, что $f(n)$ не превосходит любого элемента данного множества:

$\forall_{fMA}$(последовательность(f, A) & конечное$(M) \rightarrow \exists_n(0 \leq -f(n) + \inf($образ$(f, M))$ & n – натуральное))

Так как прием исключает несущественную неизвестную n, то имеет заголовок "связка". В случае пустоты образа M точная нижняя грань равна ∞, и можно выбрать любое значение n.

13. Нормализатор "норминф". В нормализатор общей стандартизации "норминф" включены некоторые из приведенных выше простых приемов: для объединения множеств, числовых промежутков и конечных множеств. Добавлен прием $\inf(\emptyset) = \infty$. Кроме того, введена группа приемов, позволяющих определять точную нижнюю грань образа числового множества при дифференцируемом отображении. Они аналогичны рассмотренным выше приемам нормализатора "нормнижняягрань".

7.7.7 Приемы символа "суп"

Эти приемы аналогичны приемам символа "инф".

7.7.8 Приемы символа "огрснизу"

1. Вывод посылки о существовании нижней грани. Если имеется посылка, утверждающая, что множество ограничено снизу, то выводится следствие о существовании его нижней грани:

   $\forall_a$(огрснизу$(a) \rightarrow \exists_b$(нижняягрань$(b, a)$))

   Выведенная посылка снабжается комментарием "кванторнаясвертка", блокирующим преобразование бескванторной переформулировки ("свертки").

2. Кванторная свертка. Если утверждение о существовании нижней грани не выделено комментарием "кванторнаясвертка", то оно преобразуется в утверждение об ограниченности множества снизу:

   $\forall_a$(огрснизу$(a) \leftrightarrow \exists_b$(нижняягрань$(b, a)$))

   Замена выполняется справа налево.

3. Расшифровка по определению. Если утверждение об ограниченности множества снизу является условием задачи на доказательство, то на достаточно высоких уровнях (5 либо 6) происходит его кванторная расшифровка. Это же предпринимается, если указанное утверждение является условием задачи на описание, имеющей цель "развертка". Теорема приема - та же, что в предыдущем пункте.

4. Ограниченность снизу конечного множества.

   $\forall_a$(огрснизу$\{; a\}$)

5. Проверочный оператор "усмогрснизу". Оператор имеет несколько простых приемов: усмотрение ограниченности снизу множества b из утверждения "нижняягрань(a, b)"; ограниченность снизу образа подмножества области определения ограниченной функции; ограниченность снизу множества членов сходящейся числовой последовательности.

7.7.9 Приемы символа "огрсверху"

Эти приемы аналогичны приемам символа "огрснизу".

7.7.10 Приемы, основанные на аксиоме непрерывности

Аксиома непрерывности утверждает, что для двух непустых числовых множеств, обладающих тем свойством, что каждый элемент второго - верхняя грань для первого (либо каждый элемент первого - нижняя грань для второго), существует число, являющееся одновременно нижней гранью второго множества и верхней гранью первого. В решателе аксиома непрерывности понадобилась для решения нескольких простых теоретических задач, и это повлекло за собой появление приемов вывода, основанных на данной аксиоме.

7.7.11 Вывод из двух кванторных посылок с неравенствами для точной верхней и нижней граней

Если есть две кванторных импликации, одна из которых мажорирует величину $p(x)$ точной нижней гранью множества $A(y)$, а другая - минорирует величину $q(z)$ точной верхней гранью этого же множества, то выводится кванторное неравенство, непосредственно связывающее $p(x)$ с $q(z)$:

$$\forall_{fgApmnqh}(\exists_y(g(x,y)\ \&\ n(z,y)) = h(x,u)\ \&\ \forall_{xy}(f(x)\ \&\ g(x,y) \rightarrow p(x) \leq \inf(A(y)))\ \&$$
$$\forall_{uv}(m(u)\ \&\ n(u,v) \rightarrow \sup(A(v)) \leq q(u)) \rightarrow \forall_{xz}(f(x)\ \&\ m(z)\ \&\ h(x,z) \rightarrow p(x) \leq q(z)))$$

Первый антецедент выделен указателем "идентификатор"; подкванторное выражение его левой части обрабатывается вспомогательной задачей на описание, имеющей неизвестную y. При этом сам квантор существования тоже обрабатывается вспомогательной задачей. Фильтр "элементарно(значение(x8 набор(x23 x20)))" требует, чтобы после упрощений получилось элементарное утверждение.

7.7.12 Сравнение прообразов

Пусть f - числовой набор и A, B - два непустых подмножества его области значений. Условие, что первым идет вхождение элемента множества A, переформулируется в терминах первого вхождения представителя объединения множеств A, B:

$$\forall_{fAB}(\neg(\text{прообраз}(f,A) = \emptyset)\ \&\ \neg(\text{прообраз}(f,B) = \emptyset)\ \&\ f - \text{слово} \rightarrow \inf(\text{прообраз}($$
$$f,A)) < \inf(\text{прообраз}(f,B)) \leftrightarrow f(\inf(\text{прообраз}(f,A \cup B))) \in A \setminus B)$$

7.8 Приемы символа "внутренность"

Для определения внутренности множества (вещественных чисел, числовых наборов, комплексных чисел) созданы несколько простых приемов. Кроме того, имеется нормализатор "нормвнутренность", содержащий ряд дополнительных приемов. Выражения "внутренность(...)", "замыкание(...)" в заменяющих частях приводимых ниже тождеств обрабатываются нормализаторами "нормвнутренность", "нормзамыкание". Очевидная неполнота приемов данного раздела объясняется тем, что они вводились исключительно по мере надобности, для обслуживания задач из прочих разделов.

Внутренность промежутка

$\forall_{abcd}($внутренность$([a,b]) = (a,b))$

Здесь c, d - указатели типов концов промежутка в левой части равенства.

Внутренность пересечения

$\forall_{AB}($внутренность$(A \cap B) =$ внутренность$(A) \cap$ внутренность$(B))$

Внутренность прямого произведения

$\forall_{ab}($внутренность$(a \times b) =$ внутренность$(a) \times$ внутренность$(b))$

Внутренность разности

$\forall_{ab}($внутренность$(a \setminus b) =$ внутренность$(a) \setminus$ замыкание$(b))$

Внутренность множества комплексных чисел

внтуренность$(\mathbf{C}) = \mathbf{C}$

Внутренность серии точек

$\forall_f($внутренность$(\mathrm{set}_x(\exists_n(n -$ целое $\& \ x = f(n)))) = \emptyset)$

Нормализатор "нормвнутренность"

1. Внутренность объединения.

   $\forall_{ab}($внутренность$(a \cup b) =$ внутренность$(a) \cup$ внутренность$(b))$

2. Внутренность пересечения. Так же, как для приема сканирования задачи.

3. Внутренность промежутка. Аналогично предыдущему.

4. Внутренность конечного списка.

   $\forall_a($внутренность$\{;a\} = \emptyset)$

5. Внутренность дополнения к серии.

   $\forall_f($внутренность$(\mathrm{set}_x(x -$ число $\& \ \forall_n(n -$ целое $\to \neg(x = f(n)))) = \mathrm{set}_x(x -$ число $\& \ \forall_n(n -$ целое $\to \neg(x = f(n))))))$

6. Условное выражение.

   $\forall_{abc}($внутренность$((a$ при $b,$ иначе $c)) = ($внутренность(a) при $b,$ иначе внутренность$(c)))$

7. Пустое множество.

   внутренность$(\emptyset) = \emptyset$

8. Внутренность прямого произведения. Аналогично приему сканирования задачи, однако учтены случаи двух, трех, четырех и пяти сомножителей.

9. Внутренность множества точек плоскости либо пространства, заданного системой неравенств.

   $\forall_{abfg}$(внутренность(set_z($x -$ число & $y -$ число & $a \leq y$ & $y \leq b$ & $f(y) \leq x$ & $x \leq g(y)$))) $=$ set_z(x − число & y − число & $a < y$ & $y < b$ & $f(y) < x$ & $x < g(y)$))

   Переменная z здесь выделена указателем "кортежпеременных" и идентифицируется с набором связанных переменных. Фильтры "входит(x23 x25)", "входит(x24 x25)", "длинатекста(x25 2)" означают, что этот набор представляет собой пару идущих в произвольном порядке переменных x, y. Таким образом, прием обслуживает одновременно случаи ориентации "криволинейной трапеции" вдоль оси абсцисс и вдоль оси ординат. Указатель "обобщподст(фикс(0 1 1))" обеспечивает отмену проверки непересечения переменных z с прочими термами, расположенными под $\mathrm{set}_z(\ldots)$, в частности, непересечения их с x, y. Однако, при такой отмене необходимо специально оговорить, что переменные x, y не входят в a, b. Это делает фильтр "не(контекст(список(x3 x23 x24)список(x4 x1 x2)входит(x3 x4)))". Фильтры "не(входит(x23 фикс(0 1 1 2 5 1)))". "не(входит(x23 фикс(0 1 1 2 6 2)))" проверяют невхождение переменной x в $f(y), g(y)$. Указатели "альтернатива(...)" разрешают замену любого нестрогого неравенства в левой части тождества на строгое. Кроме данного приема, для случая плоскости введены еще два. Они аналогичны только что рассмотренному, но неравенство для переменной y - единственное (в одном случае y ограничено снизу, в другом - сверху). Наконец, для трехмерного случая введен следующий прием:

   $\forall_{abfgpq}$(внутренность(set_v(x − число & y − число & z − число & $a \leq x$ & $x \leq b$ & $f(x) \leq y$ & $y \leq g(x)$ & $p(x,y) \leq z$ & $z \leq q(x,y)$))) $=$ set_v(x − число & y − число & z − число & $a < x$ $x < b$ & $f(x) < y$ & $y < g(x)$ & $p(x,y) < z$ & $z < q(x,y)$))

   Указатели и фильтры этого приема аналогичны указателям и фильтрам предыдущего.

10. Внутренность множества комплексных чисел.

    внутренность(set_z(z − комплексное)) $=$ set_z(z − комплексное)

11. Внутренность серии точек. Аналогично случаю приема сканирования задачи.

7.9 Приемы символа "арифмпрогрессия"

Несколько приемов на множество членов арифметической прогрессии возникли при обучении решателя элементарной алгебре. Ниже "арифмпрогрессия(a b)" обозначает множество членов арифметической прогрессии с начальным членом a и знаменателем b.

Усмотрение арифметической прогрессии

$\forall_{abcm}$(m − целое $\rightarrow$ set_x($\exists_n$($x = (an+b)/c$ & n − целое & $m \leq n$)) $=$ арифмпрогрессия($(am+b)/c, a/c$))

$\forall_{abcm}$(m − целое $\rightarrow$ set_x($\exists_n$($x = (an+b)/c$ & n − целое & $n \leq m$)) $=$ арифмпрогрессия($(am+b)/c, -a/c$))

Уравнения с арифметическими прогрессиями

Условие равенства двух арифметических прогрессий, имеющих одинаковые начальные члены, третьей арифметической прогрессии, расшифровывается при помощи следующей эквивалентности:

$\forall_{abcpq}(\text{арифмпрогрессия}(a,b)\cup\text{арифмпрогрессия}(a,c)=\text{арифмпрогрессия}(p,q)\leftrightarrow 0\leq bc\ \&\ p=a\ \&\ (b/c-\text{целое}\ \&\ \neg(c=0)\ \&\ q=c\vee c/b-\text{целое}\ \&\ \neg(b=0)\ \&\ q=b))$

Равенство двух арифметических прогрессий сводится к равенству их начальных членов и знаменателей:

$\forall_{abcd}(\text{арифмпрогрессия}(a,b)=\text{арифмпрогрессия}(c,d)\leftrightarrow a=c\ \&\ b=d)$

Усмотрение отличия числового множества от арифметической прогрессии

Множество, содержащее невырожденный числовой промежуток, не является арифметической прогрессией:

$\forall_{abcdefg}(0<b-a\rightarrow\neg([a,b]\cup c=\text{арифмпрогрессия}(d,e)))$

7.10 Приемы символа "граница"

В разделе содержатся: прием для определения границы отрезка, а также нормализатор "нормграница". Последний, кроме границы отрезка, определяет границу криволинейной трапеции и ее трехмерного аналога.

7.11 Приемы символа "областьграницы"

Выражение "областьграницы(A)" определяет область, ограниченную замкнутой линией A, рассматриваемой как множество точек плоскости. Это выражение встречается во многих задачах математического анализа, например, связанных с двойными интегралами. Чтобы переходить к явному описанию области в терминах неравенств, создана реализованная на ЛОСе достаточно сложная процедура, которая будет описана ниже (см. раздел, посвященный приемам алгебры множеств, реализованным на ЛОСе). Здесь же ограничимся несколькими приемами, выполняющими предварительное упрощение задания границы.

Попытка явного разрешения участка границы плоской фигуры относительно второй переменной

$\forall_{fgPab}(\text{set}_{xy}(f(x,y)=g(x,y)\ \ \&\ P(x,y))=b\rightarrow\text{областьграницы}(\text{set}_{xy}(f(x,y)=g(x,y)\ \&\ P(x,y))\cup a)=\text{областьграницы}(a\cup b))$

В антецеденте, выделенном указателем "идентификатор", предпринимается обращение к вспомогательной задаче на описание для явного разрешения утверждений $f(x,y)=g(x,y),P(x,y)$ относительно неизвестной y. После этого внешний описатель "класс" обрабатывается нормализатором "нормкласс". Проверяется, что b является объединением одного или нескольких описателей "класс", в каждом из которых либо y явно выражено через x, либо x явно выражено через параметры, отличные от x,y. Проверяется, что исходное равенство $f(x,y)=g(x,y)$ содержит y, но не является явно разрешенным относительно y. Наконец, имеются фильтры, проверяющие, что явное разрешение относительно y не чрезмерно сложно: например, что x не встречается под двумя вложенными друг в друга радикалами. Введен ограничитель трудоемкости.

Попытка усмотрения вертикальных отрезков границы

$\forall_{fgPab}(\mathrm{set}_{xy}(f(x) = g(x)\ \&\ P(x,y)) = b \to \text{областьграницы}(\mathrm{set}_{xy}(f(x) = g(x)\ \&\ P(x,y)) \cup a) = \text{областьграницы}(b \cup a))$

Первый антецедент обращается к вспомогательной задаче на описание, разрешающей утверждения $f(x) = g(x), P(x,y)$ относительно x. Внешний описатель "класс" затем обрабатывается нормализатором "нормкласс". Проверяется, что b представляет собой объединение одного или нескольких описателей "класс", фиксирующих значение x при варьируемом значении y. Проверяется также, что равенство $f(x) = g(x)$ не является явно разрешенным относительно x.

Явное разрешение условия на значения первой переменной

Если фрагмент границы явно разрешен относительно y, то предпринимается попытка разрешить остаточное условие на x:

$\forall_{rfPa}(P(x) = r \to \text{областьграницы}(\mathrm{set}_{xy}(P(x)\ \&\ y = f(x)) \cup a) = \text{областьграницы}(\mathrm{set}_{xy}(r\ \&\ y = f(x)) \cup a))$

Антцедент явно разрешает относительно x утверждение $P(x)$, обращаясь для этого к вспомогательной задаче на описание.

Ориентация равенства

Чтобы избежать громоздких кратных интегралов по области, заданной с помощью выражения "областьграницы", могут вводиться вспомогательные обозначения (переменные) для данной области. В этом случае равенство, вводящее обозначение, ориентируется так, чтобы переменная была заменяющим термом:

$\forall_{ab}(a = \text{областьграницы}(b) \leftrightarrow \text{областьграницы}(b) = a)$

Прием применяется к посылке задачи на преобразование, условие которой содержит кратный интеграл по области, задаваемой с участием переменной a. Указатель "примечание(ориентацияравенства)" закрепляет выбранную ориентацию равенства, блокируя реализованный на ЛОСе прием перестановки его частей "из общих соображений".

Отбрасывание изолированной точки

$\forall_{abc}(\text{областьграницы}(\{(a,b)\} \cup c) = \text{областьграницы}(c))$

$\forall_{abc}(\text{областьграницы}(\mathrm{set}_{xy}(x = a\ \&\ y = b) \cup c) = \text{областьграницы}(c))$

Второй прием имеет указатель "обобщподст(фикс(0 1 1 1))", разрешающий вхождения переменных x, y в выражения a, b.

Усмотрение трехмерного цилиндрического множества

Выражение "областьграницы(...)" было использовано в решателе не только для двумерного, но и для трехмерного случая. Однако, здесь рассматриваются лишь множества, ограниченные сверху и снизу поверхностями, уравнения которых явно разрешены относительно z, а сбоку - ограниченные цилиндрической поверхностью:

$\forall_{ABCabcfgP}(P(a,b) = ((a,b) \in \text{областьграницы}(\bigcup_{i=1}^{n} \mathrm{set}_{xy}(C(i)))) \,\&\, 0 \leq g(a,b)-f(a,b) \rightarrow$
$(a,b,c) \in \text{областьграницы}(\mathrm{set}_{xyz}(z = f(x,y) \,\&\, A(x,y)) \cup \mathrm{set}_{xyz}(z = g(x,y)$
$\&\, B(x,y)) \cup \bigcup_{i=1}^{n} \mathrm{set}_{xyz}(z - \text{число} \,\&\, C(i))) \leftrightarrow c - \text{число} \,\&\, f(a,b) \leq c \,\&\, c \leq g(a,b) \,\&$
$P(a,b) \,\&\, A(a,b) \,\&\, B(a,b))$

Прием преобразует условие принадлежности тройки (a,b,c) области указанного выше цилиндрического вида. Боковая поверхность, ограничивающая область, представлена здесь членом $\bigcup_{i=1}^{n} \mathrm{set}_{xyz}(z - \text{число} \,\&\, C(i))$. Указатели "содержится(x23 x28)", "содержится(x24 x28)" разрешают вхождения переменных x, y в выражения $C(i)$. Указатель "развертка(фикс(0 1 2 1 3)фикс(1 2 2 1))" определяет идентификацию и запись рассматриваемых в теореме приема конечных объединений как обычных многоместных объединений. Первый антецедент преобразует условие принадлежности пары (a,b) области, ограниченной проекцией боковой поверхности на плоскость Oxy. Для этого используется обращение к вспомогательной задаче на преобразование, при решении которой работает обычная "двумерная" техника нахождения области с заданной границей. $P(a,b)$ есть результирующее условие на a, b. Второй антецедент проверяет, что верхняя ограничивающая поверхность не ниже нижней. В качестве дополнительных посылок он использует утверждения $A(x,y), B(x,y), P(x,y)$, определяющие проекцию области на плоскость Oxy. Правая часть эквивалентности состоит из неравенств для третьей кординаты и условий на проекцию области.

Попытка явного разрешения участка границы трехмерной области относительно третьей переменной

Этот прием подготавливает возможность применения предыдущего приема, разрешая относительно третьей переменной уравнения ограничивающих поверхностей:

$\forall_{fgPab}(\mathrm{set}_{xyz}(f(x,y,z) = g(x,y,z) \,\&\, P(x,y,z)) = b \rightarrow \text{областьграницы}(\mathrm{set}_{xyz}(f(x,$
$y,z) = g(x,y,z) \,\&\, P(x,y,z)) \cup a) = \text{областьграницы}(a \cup b))$

Указатели и фильтры аналогичны случаю разобранного выше приема для явного разрешения участка границы плоской фигуры относительно второй переменной.

Усмотрение пустой области

Считаем, что единственная самонепересекающаяся линия ограничивает пустую область:

$\forall_{fgh}(\text{областьграницы}(\mathrm{set}_{xy}(y = f(x) \,\&\, P(x))) = \emptyset)$

7.12 Приемы символа "числовойотрезок"

Утверждение "числовойотрезок(a)" означает, что a - непустой отрезок числовой прямой (концы в отрезок включаются; допускается случай их совпадения друг с другом).

Принадлежность пересечению семейства числовых отрезков

Условие принадлежности точки пересечению семейства числовых отрезков переформулируется в виде конъюнкции двух условий - эта точка является верхней гранью для множества левых концов отрезков и нижней гранью для множества правых концов:

$\forall_{ABa}$(числовойотрезок$(B(i)) \to a \in \bigcap_{i,A(i)} B(i) \leftrightarrow$ верхняягрань$(a, \mathrm{set}_x(\exists_i(A(i)$ & $x = \inf(B(i)))))$ & нижняягрань$(a, \mathrm{set}_x(\exists_i(A(i)$ & $x = \sup(B(i))))))$

Вложенные числовые отрезки

Если имеется посылка, явно указывающая, что A - последовательность вложенных множеств, причем усматривается, что эти множества суть числовые отрезки, то выводятся кванторные импликации о неубывании левых концов отрезков и невозрастании правых концов:

$\forall_A$(убывмножества(A) & числовойотрезок$(A(i)) \to \forall_{ij}(i \in \mathbf{N}$ & $j \in \mathbf{N}$ & $i \leq j \to \inf(A(i)) \leq \inf(A(j)))$ & $\forall_{ij}(i \in \mathbf{N}$ & $j \in \mathbf{N}$ & $i \leq j \to \sup(A(j)) \leq \sup(A(i))))$

Прием применяется в задачах на описание, имеющих цель "пример", так как новые кванторные импликации могут оказаться полезными для вывода следствий, дающих искомый пример.

Проверочный оператор "усмчисловойотрезок"

Специальных приемов для усмотрения числового отрезка всего два. Первый прием относится к множествам, заданным в виде числового промежутка. Здесь проверяется неотрицательность разности концов. Второй прием ищет в посылках кванторную импликацию вида $\forall_i(A(i) \to$ числовойотрезок$(B(i)))$ и пытается ее использовать.

7.13 Приемы символа "замыкание"

В разделе представлен лишь нормализатор "нормзамыкание". Он имеет приемы для определения замыкания простейших множеств: конечного промежутка, полуоси, всей вещественной оси, а также уже замкнутого множества. Свойство замкнутости распознается здесь с помощью оператора "усмзамкнутое" (см. следующий пункт).

7.14 Приемы символа "замкнутое"

В разделе представлен лишь проверочный оператор "усмзамкнутое". Он усматривает замкнутость числовых промежутков и конечных множеств; предпринимает попытку установить замкнутость объединения конечного числа множеств путем доказательства замкнутости каждого множества по отдельности; усматривает замкнутость множества комплексных чисел, имеющего вид прямого произведения подмножеств вещественной и мнимой осей.

7.15 Упражнения

Создать приемы, необходимые для решения следующих задач:

1. Упростить выражение $\mathbf{Q} \cap \mathbf{Z}$.
2. При заданном вещественном a найти все такие множества x, для которых a является верхней гранью.
3. Вычислить точную нижнюю грань множества $\mathrm{set}_x(\exists_y(x = \sin(y)$ & $y -$ число$))$.
4. Найти точную верхнюю грань множества $\mathrm{set}_m(\exists_n(n-$целое & $m = 1+n-2n^2))$. Найти наибольший элемент этого множества.
5. Найти все такие вещественные x, для которых множество $\mathrm{set}_n(\exists_m(m$ – натуральное & $n = xm))$ ограничено снизу.
6. Найти внутренность круга $\mathrm{set}_{xy}(x^2 + y^2 \leq 1)$.
7. Найти границу того же круга.
8. Найти замыкание множества $\mathrm{set}_x(\exists_n(n$ – натуральное & $x = 1/n))$.
9. Найти все предельные точки того же множества.
10. Создать проверочный оператор "усмОбласть" для усмотрения связных открытых множеств.

7.15.1 Указания

1. Вводим задачу на преобразование, имеющую условие $\mathbf{Q} \cap \mathbf{Z}$, и запускаем ее решение. Обнаруживаем, что выражение не упрощается. Мы знаем, что имеется прием упрощения пересечения двух множеств, одно из которых включается в другое. Раз он не применяется, значит, отсутствуют средства проверки включения множества целых чисел в множество рациональных чисел. Находим проверочный оператор "усмсодержится". В нем не оказывается никаких приемов для усмотрения подмножеств множества рациональных чисел. Поэтому можно ввести несколько таких приемов, хотя бы для основных случаев - множеств натуральных, целых неотрицательных и целых чисел. Для нашего примера создаем прием с теоремой $\mathbf{Z} \subseteq \mathbf{Q}$ и заголовком "спуск(усмсодержится)". Единственный его фильтр - "уровень(1)". Рекомендуется самостоятельно создать несколько более общие приемы - например, для усмотрения включения в множество рациональных чисел множеств, заданных описателями вида $\mathrm{set}_x(x - P$ & $Q(x))$, где P - один из символов "натуральное", "целое", "рациональное".

2. Вводим задачу на описание, имеющую посылку "a – число" и условия "верхняягрань(a, x)", "$x \subseteq \mathbf{R}$". Неизвестной служит переменная x. Эта задача системой не решается. Находим подраздел базы приемов, относящийся к верхним граням числовых множеств. В нем нет никаких приемов, позволяющих разрешать предикат "верхняягрань(a, x)" относительно x. Вводим для этой цели следующий прием: $\forall_{ax}(a-$число & $x \subseteq \mathbf{R} \rightarrow$ верхняягрань$(a, x) \leftrightarrow x \subseteq (-\infty, a])$. Напоминаем, что круглая и квадратная скобки при задании числового промежутка вводятся нажатиями клавиш "Ctrl- скобка". Фильтры приема задаем, например, такие: "тип(описать)", "условие", "уровень(2)", не(известно(x23))", "известно(x1)". Оба антецедента выделяем указателями "блокпроверок".

3. Вводим задачу на упрощение выражения $\inf(\mathrm{set}_x(\exists_y(x = \sin(y)\ \&\ y - \text{число})))$ и запускаем решатель. Он никак не изменяет данное выражение. Переходим в раздел базы приемов, относящийся к символу "инф", и анализируем имеющиеся приемы. Оказывается, что подходящих приемов для задач рассматриваемого типа там нет. Имеется несколько возможностей для восполнения пробела. Например, можно было бы создать прием, переформулирующий задачу в терминах образов числовых множеств и воспользоваться приемами нормализатора "норминф". Рассмотрим другой подход - создадим прием, предпринимающий попытку разрешить подкванторное условие относительно связанной переменной квантора существования. Такое разрешение сделает вероятным последующее исключение квантора. Теорема приема имеет вид: $\forall_{PQ}(P(x,y) = Q(x,y) \to \inf(\mathrm{set}_x(\exists_y(P(x,y)))) = \inf(\mathrm{set}_x(\exists_y(Q(x,y)))))$. Указатель "отображение(x40 x41)" делает переменные P, Q функциональными. Первый антецедент, выделенный указателем "идентификатор(1)", обращается к вспомогательной задаче на разрешение условия $P(x,y)$ относительно переменной y. Его левая часть обрабатывается нормализатором "задача(5 тип(описать)полный явное прямойответ упростить цель(неизвестная(x24)))". Уровень срабатывания приема выбираем равным 4, чтобы в простых случаях успевали сработать другие приемы определения точной нижней грани. После компиляции приема решатель выдает ответ "-1".

4. Вводим задачу на упрощение выражения $\sup(\mathrm{set}_m(\exists_n(n - \text{целое}\ \&\ m = 1 + n - 2n^2)))$. Решатель не получает требуемый ответ, даже если создать прием для sup, аналогичный приему для inf из предыдущей задачи. Переходим в раздел базы приемов, соответствующий символу "суп". В нем находим подраздел "числовая последовательность". Выбираем пункт "анализ локальных максимумов и верхнего предела". Здесь имеется прием, определяющий точную верхнюю грань числовой последовательности путем поиска точек локального максимума и вычисления верхнего предела последовательности. Однако, в нашей задаче параметр n не натуральный, а целочисленный. Поэтому придется создать альтернативную версию найденного приема, подходящую для целочисленного параметра. Чтобы ускорить процесс, скопируем сначала старую версию (Ctr-д и F4). Выделим подутверждение теоремы "$n -$ натуральное", нажмем клавишу "Ф" и введем альтернативную версию "$n -$ целое". Повторим это для каждого из двух вхождений в теорему. Затем выделим подутверждение "$n = 1$" и нажмем "Ctrl-Del" для его удаления. Так же удалим "$2 \leq n$". Чтобы рассмотреть верхний предел в минус-бесконечности, добавим антецедент $q = \overline{\lim}_{n\to\infty}\{f(-n)\}$. Этого можно добиться, выделив последний антецедент старой версии, нажав "В" и введя формульным редактором указанный дополнительный антецедент. Наконец, заменим в старой версии теоремы подвыражение $\{p\}$ на $\{p, q\}$. Используя текстовый редактор, пополняем кванторную приставку символом q (x16). Далее корректируем указатель "идентификатор(1 2)", заменяя его на "идентификатор(1 2 3)", и обрабатываем нормализатором "задача(5 упростить)" правую часть равенства для q. Компилируем прием и запускаем решатель. Теперь получается ответ 1.

5. Вводим задачу на описание, имеющую условие "огрснизу$(\mathrm{set}_n(\exists_m(m -$ натуральное $\&\ n = mx)))$" и единственную неизвестную x. Запускаем ее решение и обнаруживаем, что решатель "зависает" - не выдает ни ответа, ни отказа. Чтобы выяснить причину, нажимаем "Break", затем - пробел (для выхода на

уровень трассировки по срабатываниям приемов), и "Enter" - для начала такой трассировки. Нажимая далее многократно "Enter", легко выявляем зацикливание: сначала происходит исключение квантора существования путем явного разрешения подкванторного условия относительно m, а затем снова предпринимается ввод вспомогательного целочисленного параметра m вместе с квантором существования. Чтобы устранить зацикливание, нужно заблокировать хотя бы одно их этих действий. Видимо, естественно отказаться от попытки исключения только что введенного параметра m. Чтобы найти способ заблокировать эту попытку, переходим к просмотру ЛОС-программы приема "Разрешение подкванторного условия относительно переменных кванторной приставки". Для этого в момент срабатывания приема нажимаем "End", из главного меню входим в оглавление программ, и далее нажимаем "курсор вправо". Анализируем начало программы приема, где расположены различные фильтры, блокирующие его срабатывание. В частности, обнаруживаем фильтр "коммент(x1 связприставка подтерм(x2))", который подсказывает способ блокировки: ввод комментария (связприставка A) для рассматриваемого квантора существования A. Можно просмотреть информацию о символе "связприставка" и узнать, что такой комментарий свидетельствует о ранее предпринимавшейся попытке разрешения A. Далее переходим к оглавлению приемов, и в разделе ("Описатель КЛАСС", "Параметрическое описание класса", "Ввод вспомогательного целочисленного параметра") находим упомянутый выше прием, вводящий целочисленный параметр m. Добавляем к нему указатель "замечание(связприставка фикс(0 2 2))", перекомпилируем (F3) и повторно запускаем решение. Теперь зацикливание пропадает, но ответ не находится.

Очевидно, нужно создать прием, выполняющий расшифровку условия ограниченности снизу множества, заданного описателем "класс". Вводим прием по следующей теореме:

$\forall_P(\text{огрснизу}(\text{set}_x(P(x))) \leftrightarrow \exists_y(y - \text{число} \ \& \ \forall_x(P(x) \to y < x)))$

Сопровождаем прием фильтрами "уровень(2)", "условие", "тип(описать)", "корень", "не(известно(корень))". Нажимаем "н", и получаем следующие подсказанные генератором приемов указатели: "кортежпеременных(x23)", "отображение(x40)". Компилируем прием, запускаем решение задачи и получаем ответ "$x - \text{число}, 0 \leq x$".

6. Вводим задачу на преобразование выражения "внутренность($\text{set}_{xy}(x^2 + y^2 \leq 1)$)". Запускаем решение, и обнаруживаем, что утверждение под описателем "класс" разрешено относительно переменных связывающей приставки явным образом, однако внутренность не найдена. Переходим в раздел базы приемов, связанный с символом "внутренность", для рассмотрения уже введенных ранее приемов. В нормализаторе "нормвнутренность" обнаруживаем прием для определения внутренности множества точек плоскости, заданного системой неравенств, явно разрешенной относительно координат. Берем этот прием (выбираем из нескольких представленных в разделе подходящий для нашего случая, т.е. последний) и копируем его нажатием "Ctr-д". Заменяем заголовок на "втойтерм" и добавляем фильтр "уровень(2)". Компилируем прием и запускаем решатель. Обнаруживаем, что после применения приема начинается цикл совершенно ненужных попыток явно разрешить утверждение под описателем "класс". Анализируя прием, предпринимающий такую попытку, замечаем в нем

указатель "ключ(класс теквхожд)", блокирующий срабатывание при наличии комментария (класс A), где A - описатель "класс". Поэтому вводим в нашем новом приеме указатель "замечание(класс фикс(0 2))". Запускаем решатель и обнаруживаем, что блокировка не состоялась. Изучаем причину этого с помощью отладчика. Оказывается, что наш прием применяет к заменяющему терму, внутри оператора "заменавхождения", простейшее переупорядочение операндов, а в комментарии (класс ...) этого переупорядочения нет. В результате при последующий преобразованиях комментарий (класс ...) не корректируется синхронно с изменениями под описателем, и к моменту попытки разрешения наблюдается полное рассогласование между текущим выражением и блокирующим комментарием. Чтобы исправить рассогласование, обрабатываем в новом приеме заменяющий терм нормализатором "стандупорядочение". Компилируем прием, запускаем решатель и получаем ответ.

Рекомендуется самостоятельно создать такую версию решения данной задачи, при которой ответ не был явно разрешен относительно переменных x, y.

7. Пример аналогичен предыдущему. Нужно использовать копию уже имеющегося в нормализаторе "нормграница" приема, преобразовав ее в прием сканирования задачи. Указатели "замечание(класс ...)" при этом относятся к каждому из четырех классов заменяющего выражения, причем все они должны быть обработаны нормализатором "стандупорядочение".

8. Чтобы решить эту задачу, проще всего воспользоваться аппаратом нахождения множества всех частичных пределов последовательности, представленным (хотя и минимальным образом) в разделе базы приемов ("Математический анализ", "Предел", "Нахождение частичных пределов"). Собственно говоря, нет необходимости знакомиться с данным аппаратом - достаточно лишь знать о его существовании. Утверждение о том, что число a является частичным пределом последовательности f, записывается в решателе как "частичныйпредел(a f)". Вводим новый прием, позволяющий находить замыкание множества членов числовой последовательности путем добавления к нему всех конечных частичных пределов:

   $\forall_{fPx}((\text{частичныйпредел}(x, \lambda_n(f(n), n - \text{натуральное})) \ \& \ x - \text{число}) = P(x) \to \text{замыкание}(\text{set}_y(\exists_n(n - \text{натуральное} \ \& \ y = f(n)))) = \text{set}_y(\exists_n(n - \text{натуральное} \ \& \ y = f(n))) \cup \text{set}_y(P(y)))$

   Единственный антецедент выделен указателем "идентификатор". Он обращается к вспомогательной задаче на описание для нахождения утверждения $P(x)$, определяющего искомые конечные частичные пределы x. Левая часть антецедента сопровождается нормализатором "задача(5 тип(описать) полный явное прямойответ цель(неизвестная(x23)))". Указатель "новаяпеременная(x23)" вводит вспомогательную переменную x.

9. Пример аналогичен предыдущему: снова используется описание множества частичных пределов последовательности.

10. Входим в раздел "Область" оглавления базы приемов и нажимаем "Ctrl-п". Вводим название оператор "усмОбласть" и выбираем пункт "Проверочный оператор". Нажимаем левую кнопку мыши в окне "Вид проверяемого утверждения" и набираем текстовым редактором "Область(x1)". После нажатия "Enter"

входим в окно "Упорядочение входных переменных" и вводим формульным редактором "a". Число уровней срабатывания полагаем равным 3. Наконец, выбираем окно "Ввести". Рекомендуется самостоятельно создать приемы для усмотрения одномерной области (открытого промежутка) и двумерной области вида $\mathrm{set}_{xy}(a < x \ \& \ x < b \ \& \ f(x) < y \ \& \ y < g(x))$. Затем создать прием сканирования задачи, обращающийся к проверочному оператору "усмОбласть", ввести тестовую задачу на доказательство утверждения вида "Область(A)", и проверить правильность работы созданных приемов.

Глава 8

Приемы для множеств и функций, реализованные на ЛОСе

Перейдем к рассмотрению небольшой группы связанных с множествами и функциями приемов, которые оказалось удобно реализовать на ЛОСе. Эти приемы могут быть найдены в разделе ("Приемы решателя", "Алгебра множеств") оглавления программ.

Мощность класса наборов, условие принадлежности которому распадается на компоненты связности

Прием срабатывает на уровне 1 при обнаружении символа "мощность" в условии задачи на преобразование. Если этот символ является корнем выражения $\mathrm{card}(\mathrm{set}_x A)$, причем связывающая приставка x имеет не менее двух переменных, то предпринимается попытка представить A в виде A_1 & A_2, где A_1 зависит от одних переменных списка x, а A_2 - от других. Затем рассматриваемое выражение заменяется на произведение выражений $\mathrm{card}(\mathrm{set}_y A_1)$, $\mathrm{card}(\mathrm{set}_z A_2)$. Здесь y, z - разбиение списка x, определяемое выбором выражений A_1, A_2.

Усмотрение функциональности множества наборов

Напомним, что "функционально(A)" - утверждение, означающее, что A есть множество наборов одинаковой длины, являющееся графиком функции. Оно не содержит двух различных наборов, у которых совпадают все соответствующие разряды, кроме последних. Прием усматривает условие "функционально(класс$(x_1 \dots x_n P(x_1 \dots x_n))$)" задачи на доказательство и обращается к вспомогательной задаче на описание с неизвестной x_n и условием $P(x_1 \dots x_n)$. Если получается ответ, в котором x_n явно выражается через $x_1, \dots, x_{n-1}$, то условие задачи заменяется на константу "истина". Заметим, что вводимая приемом задача на описание имеет дополнительную цель "функционально", разрешающую применение различных не совсем "явных" способов выражения x_n через параметры, в обычных случаях не используемых.

Усмотрение непересечения двух классов объектов, различие типов которых усматривается при помощи справочника "характеристика"

Иногда группа одноместных предикатных символов $A_1, \dots, A_n$ используется для указания альтернативных значений некоторого свойства f. Например, предикаты "синий(x)", "красный(x)", "зеленый(x)", и т.п., - соответствуют различным взаимоисключающим значениям свойства "цвет(x)". Чтобы связывать такие предикатные символы $A_1, \dots, A_n$ со свойством f, используется запись "типы(f $A_1 \dots A_n$)". Более точно, данная запись означает, что одноместный функциональный символ f может принимать в качестве своих значений только взаимоисключающие одноместные предикатные символы $A_1, \dots, A_n$, причем равенство $f(x) = A_i$ эквивалентно истинности утверждения $A_i(x)$. Для нахождения свойства f по предикату A_i служит справочник "характеристика".

Рассматриваемый прием относится к проверочному оператору "усмнепересек". Если анализируются множества $\mathrm{set}_x(P(x)\ \&\ \dots)$ и $\mathrm{set}_y(Q(y)\ \&\ \dots)$, где P, Q - различны, то для них предпринимается обращение к справочнику "характеристика". При получении одного и того же ненулевого результата f проверочный оператор усматривает непересечение множеств.

Попытка установления равенства мощности множества объектов мощности множества параметров, через которые эти объекты выражаются

Прием усматривает вхождение выражения

$$\mathrm{card}(\mathrm{set}_{x_1 \dots x_n} \exists_{y_1 \dots y_m}(A(y_1 \dots y_m)\ \&\ B(x_1 \dots x_n y_1 \dots y_m))),$$

где $A(y_1 \dots y_m)$ идентифицируется путем группировки всех конъюнктивных членов, не зависящих от $x_1, \dots, x_n$. Путем обращений к вспомогательным задачам проверяется, что значения $y_1, \dots y_m$ образуют взаимно-однозначную параметризацию значений $x_1, \dots, x_n$. Сначала решается задача на проверку различия наборов $x_1, \dots, x_n$, определенных для различных значений $y_1, \dots, y_m$. Затем - задача на проверку существования значений $x_1, \dots x_n$ для произвольных удовлетворяющих условию A значений $y_1, ldots, y_m$. Наконец, проверяется, что значения $x_1, \dots, x_n$ по значениям $y_1, \dots, y_m$ определяются однозначно. В заключение рассматриваемое выражение заменяется на $\mathrm{card}(\mathrm{set}_{y_1 \dots y_m} A(y_1 \dots y_m))$. Прием срабатывает на уровне 5.

Склейка нескольких кванторных импликаций, определяющих значения выражения для различных подклассов разбиения

Прием инициируется при усмотрении посылки задачи на исследование, имеющей вид "разбиение(A $\{B_1, \dots, B_n\}$)". Предпринимается поиск посылок вида $\forall_x(x \in B_i \to t(x) = p_i(x))$ $(i = 1, \dots, n)$. При идентификации учитывается возможность переобозначения связанной переменной x. Затем указанные кванторные импликации заменяются на одну импликацию вида

$$\forall_x(x \in A \to t(x) = (p_1(x) \text{ при } x \in B_1, \text{иначе}(p_2(x) \text{ при } x \in B_2, \text{...иначе } p_n(x)))).$$

Уровень срабатывания приема равен 3.

Переход от суммирования по всевозможным размещениям заданной длины, не большей 4, к кратному суммированию по элементам

Усматривается выражение вида:

$$\sum_{x, \text{размещение}(x,n,A)} t(x(1), \dots, x(n)).$$

Здесь n идентифицировано с натуральной константой, не превосходящей 4; выражение $t(\dots)$ содержит переменную x только внутри подвыражений $x(1), \dots, x(n)$. Выбираются новые переменные $y_1, \dots, y_n$, и рассматриваемая сумма заменяется на кратную сумму следующего вида:

$$\sum_{y_1, y_1 \in A} \sum_{y_2, y_2 \in A, y_2 \neq y_1} \dots \sum_{y_n, y_n \in A, y_n \neq y_1, \dots, y_n \neq y_{n-1}} t(y_1 \dots y_n)$$

Уровень срабатывания приема равен 4.

Усмотрение возможности декомпозиции неизвестной последовательности на несколько подпоследовательностей

Если задача на описание имеет условие "последовательность(x A)", где x - неизвестная, и не требуется получить полное описание, то находятся все условия $f_1, \dots, f_n$, содержащие x. Пусть $n \geq 2$, причем каждое f_i имеет единственное вхождение переменной x, расположенное внутри подутверждения вида $\exists_j(j$ – натуральное & $a_i < j$ & $B_i(\dots x(j) \dots))$; $i = 1, \dots, n$ (допускаются нестрогие неравенства). Тогда можно каждое условие f_i пытаться реализовать для отдельной неизвестной последовательности y_i, а затем склеить их в общую последовательность x. Для этой цели выбираются новые переменные $y_1, \dots, y_n, k$, и решается вспомогательная задача на описание, полученная заменой условий $f_1, \dots, f_n$ на утверждения "последовательность(y_i, A)", "функция(y_i)", "F_i", "$x =$ кортежи($\lambda_k((y_1(k), \dots, y_n(k)), k -$ натуральное))" ($i = 1, \dots, n$). Здесь F_i - результат замены в f_i подвыражения $x(j)$ на $y_i(j)$. Уровень срабатывания приема равен 3.

Определение области, заданной через свои граничные линии

Описываемый в этом разделе прием часто срабатывает при решении задач по математическому анализу, в которых какая-либо плоская область задается путем указания ограничивающих ее линий. Используется обозначаение "областьграницы(A)", где A - объединение множеств точек ограничивающих линий. Прием выполняет переход к явному заданию области в терминах неравенств для координат точек.

Будем прослеживать действия, выполняемые программой приема. Выйти на нее можно через пункт ("Приемы решателя" - "Алгебра множеств" - "Определение области, заданной через свои граничные линии") оглавления программ.

Рассматривается условие задачи на преобразование, содержащее подвыражение "областьграницы(A)". Выражение A представляется в виде $A_1 \cup \dots \cup A_n$, $n \geq 1$, и переменной x6 присваивается набор выражений $A_1, \dots, A_n$. Проверяется, что все эти выражения имеют своим заголовком описатель "класс". Их связанные переменные переобозначаются так, чтобы у различных описателей они совпадали. Переменной x7 присваивается набор этих связанных переменных. Проверяется, что его длина равна 2 (случай плоскости); переменной x8 присваивается обозначающая абсциссу переменная x, переменной x9 - обозначающая ординату переменная y.

Реализуется цикл заполнения накопителя x10 ссылок на атомарные фрагменты граничных линий (см. контрольную точку "прием(2)"). Каждый такой фрагмент - либо участок графика зависимости $y = f(x)$, либо отрезок вертикальной линии $x = a$. В наборе x10 перечисляются четверки (B_1, B_2, B_3, B_4), где B_1 - переменная, относительно которой разрешен фрагмент (y в первом случае и x во втором); B_2 - выражение для переменной B_1 (соответственно, $f(x)$ либо a); B_3 - условия на противоположную переменную, ограничивающие фрагмент; B_4 - накопитель ссылок на точки пересечения фрагмента с другими фрагментами. Первоначально накопители B_4 суть символы "пустоеслово"; заполняться они будут в следующем цикле. Заметим, что атомарный фрагмент может пересекаться с другими атомарными фрагментами в произвольных точках, не обязательно концевых.

Далее (контрольная точка "прием(3)") реализуется цикл заполнения накопителя x11 указателей на точки пересечения атомарных фрагментов. Эти указатели суть пары (абсцисса и ордината точки пересечения - набор ссылок на атомарные фрагменты, к которым относится точка). Ссылки на фрагменты - левые края представляющих их четверок из накопителя x10. Элементы набора x11 упорядочены по неубыванию абсцисс точек. При определении координат точек и при упорядочении решаются вспомогательные задачи. Они сопровождаются ограничителями трудоемкости, но не слишком сильными. Даже на проверки неравенств при упорядочении точек отводится примерно по 2 млн. шагов интерпретатора, а на определение координат - порядка 32 млн. шагов. Если не удается определить все точки пересечения и упорядочить их, выполнение приема блокируется. По завершении заполнения накопителя x11 создаются необходимые ссылки на его элементы из накопителя x10.

После контрольной точки "прием(14)" предпринимается отбрасывание "висячих" фрагментов, имеющих пересечение с другими фрагментами в единственной точке. Они удаляются из списка x10, и корректируются встречные ссылки на них из списка x11. Если остается менее двух фрагментов, то область полагается пустой.

После контрольной точки "прием(13)" идет попытка разбить области на компоненты связности. Допускается наличие единичной линии, соединяющей компоненту связности с какой-либо другой компонентой. Такая линия сразу же присваивается переменной x14, затем выбирается какая-либо другая линия x15, которая дополняется до компоненты связности x16 без учета линии x14. Переменная x13 играет роль накопителя групп линий, отнесенных к различным компонентам связности. Если число компонент оказалось более одной, то прием сразу же заменяет преобразуемое выражение "областьграницы(...)" на объединение выражений аналогичного вида, соответствующих найденным компонентам. Они будут анализироваться при повторных попытках применения данного приема.

Если число компонент связности было равно 1, то после контрольной точки "прием(15)" предпринимается еще одна попытка разбить область на компоненты связности - путем обнаружения точки сочленения P. Все линии, имеющие точку с абсциссой, равной абсциссе точки P, проходят через P. В этом случае область разбивается на две части: слева от P и справа от P.

Предпринимается просмотр несамопересекающихся маршрутов, начинающихся с первого фрагмента списка x10, до обнаружения замкнутого маршрута, составленного из всех атомарных фрагментов списка x10. Таким образом выделяется область, граница которой составлена из фрагментов всех перечисленных в выражении "областьграницы(...)" линий. Перебор маршрутов ведется по принципу "сначала вглубь". Если указанного маршрута нет, прием не применяется. В противном случае переменной x16 оказывается присвоен список ссылок на точки, последовательно проходимые при перемещении вдоль маршрута (первая точка совпадает с последней, и других совпадений точек нет). Переменной x17 присваивается набор ссылок на атомарные фрагменты, последовательно соединяющие точки набора x16.

После контрольной точки "прием(6)" переменной x18 присваивается упорядоченный по неубыванию набор абсцисс точек набора x16. Этот набор определяет на оси абсцисс последовательность идущих друг за другом отрезков. Переменной x19 присваивается накопитель, заполняемый четверками (абсцисса начала очередного отрезка оси абсцисс - абсцисса конца отрезка - набор троек (атомарный фрагмент списка x10, определенный на данном отрезке и отнесенный к циклу - ордината его левого конца - ордината его правого конца) - пустой накопитель, заполняемый далее при упорядочении кривых отрезка "по высоте"). Если атомарный фрагмент продолжается в цикле через левый либо правый конец отрезка, то вместо ординаты его (представленной в формате терма) помещается логический символ 0. Вертикальные отрезки в наборе x19 не учитываются. После заполнения накопителя x19 - выход на контрольную точку "прием(7)".

Здесь выполняется цикл сравнений кривых, отнесенных в накопителе x19 к одному и тому же отрезку, "по высоте". Соответствующий накопитель (последний разряд четверки набора x19) заполняется парами (T_1, T_2), где T_i - тройки, представляющие сравниваемые на отрезке кривые, причем удается усмотреть расположение кривой T_2 над кривой T_1. Для усмотрения используются вспомогательные задачи на доказательство неравенств для разности ординат в концевой точке. Рассматриваются лишь те концы, для которых обе тройки T_1, T_2 определяют значение ординаты. Чтобы упростить анализ, ордината вычисляется не в концевой точке, а в варьируемой внутренней точке, для которой вводится дополнительная посылка о стремлении ее к концу отрезка.

После контрольной точки "прием(8)" реализуется еще один цикл для уточнения расположения "по высоте" тех кривых, которые не удалось сравнить в предыдущем цикле. Чтобы уточнить расположение двух кривых, заданных на некотором отрезке, рассматриваются продолжающие их кривые соседнего отрезка. Соотношение "по высоте" между последними переносится в накопитель текущей четверки набора x19. Как только в процессе рассмотрения четверки набора x19 удается определить взаимное расположение всех ее кривых, происходит занесение нуля в набор, являющийся последним разрядом этой четверки. По завершении цикла проверяется, что нули имеются во всех указанных наборах, т.е. определилось взаимное расположение "по высоте" для всех пар кривых.

После контрольной точки "прием(9)" реализуется цикл упорядочения кривых "по высоте". Третий элемент произвольной четверки набора x19 содержит список троек, ссылающихся на кривые текущего отрезка оси абсцисс. Эти тройки располагаются по увеличению высоты кривой.

Наконец, после контрольной точки "прием(10)" реализуется цикл заполнения накопителя x21 выражениями "класс$(xy\ldots)$" для криволинейных трапеций, соответствующих четверкам набора x19. Каждая четверка имеет четное число кривых на своем отрезке, так как эти кривые входят в общий цикл x17. Кривые разбиваются на пары, и между каждой парой рассматривается своя криволинейная трапеция. Таким образом, на одном и том же отрезке криволинейных трапеций может оказаться несколько. По завершении цикла - переход через контрольную точку "прием(11)", формирование заменяющего терма x22 для объединения криволинейных трапеций, и выполнение замены.

Замена переменных в условии принадлежности области, заданной через свои граничные линии

Прием используется при замене переменных в кратных интегралах. Текущая задача на преобразование имеет комментарий (двойнаязамена ...), хранящий выражения для новых координат через старые, а также дополнительные ограничения на новые координаты. В условии задачи встречается условие принадлежности области, заданной через свои граничные линии. Оно преобразуется к новым координатам.

Глава 9

Процедуры элементарной алгебры, реализованные на ЛОСе

Управление логическими процессами в элементарной алгебре, по сравнению с алгеброй множеств, оказалось на порядок более серьезной задачей. Можно выделить несколько особенностей этого раздела, существенно усложняющих ситуацию. Во-первых, интенсивное использование условий на область допустимых значений операций при проверке корректности преобразований. Оно требует создания специальной техники "сопровождения по о.д.з.", обеспечивающей наличие в контексте преобразуемых выражений всей необходимой для быстрых проверок информации. Впрочем, следует заметить, что не только в алгебре множеств, но даже и в "надстройках" над элементарной алгеброй - в математическом анализе, аналитической геометрии, дифференциальных уравнениях и др., - сопровождение по о.д.з. какого - либо дальнейшего развития не получило.

Особенно много хлопот с сопровождением доставляют степенные выражения, где о.д.з. оказалась составлена из нескольких кусков: положительное основание степени; рациональный неотрицательный показатель степени, имеющий нечетный знаменатель; неотрицательное основание степени и положительный рациональный показатель с четным знаменателем; ненулевое основание степени и рациональный отрицательный показатель с нечетным знаменателем. Иногда приходится дублировать приемы, ориентируя их на каждый из таких случаев.

Другая особенность элементарной алгебры - появление вычислений с константными выражениями. Здесь пришлось воспользоваться серией вспомогательных процедур, реализованных на ЛОСе. Эти процедуры были перечислены в главе 6 первого тома монографии (раздел 6.4 - "Арифметические операторы").

Следующая особенность - усложнение идентификации. Например, для идентификации произведения ab^3 как "части" произведения a^3b^5 нужна своя вспомогательная процедура - формальное сравнение множеств операндов, достаточное для алгебры множеств, здесь не годится. Аналогично, нужна особая идентификация для степенных выражений. Такая идентификация осложняется необходимостью учета контекста. Например, если в преобразуемом выражении имеется $\sqrt{x}$, то выражение x можно идентифицировать как полный квадрат; если же никаких намеков на использование в задаче радикалов нет, то данная идентификация не выполняется. Наличие указанных особенностей идентификации потребовало включения в компилятор ГЕНОЛОГа ряда специальных фрагментов, практически не используемых приемами других разделов. Возникли вспомогательные процедуры ЛОСа и комментарии задач, обслуживающие специфику элементарной алгебры. Поэтому, для лучшего понимания того, как должны работать приемы элементарной алгебры, реализованные на ГЕНОЛОГе, мы начнем с описания "подводной части" решателя - вспомогательных процедур и приемов элементарной алгебры, реализованных на ЛОСе.

9.1 Определение о.д.з. для степени

Для всех операций и отношений элементарной алгебры, кроме степени, о.д.з. находится с помощью приема, заданного на ГЕНОЛОГе. В случае степени используется процедура, реализованная непосредственно на ЛОСе. Ее программа, относящаяся к символу "степень", легко может быть найдена по своему первому оператору "обращение(одз)". Эту программу также можно найти через оглавление программ в разделе ("Приемы решателя", "Элементарная алгебра", "О.д.з. для степени"). Входные переменные суть: x1 - задача, в которой встречается степенное выражение; (x2,x3,x4) - координата вхождения в задачу этого выражения. Для степенного выражения a^b результирующий список утверждений определяется следующим образом.

Прежде всего, рассматривается случай, когда показателем степени b служит числовая константа либо числовая дробь. Находится представление b в виде несократимой простой дроби с числителем x5 и положительным знаменателем x6. Вводится накопитель результата x8. В него заносятся утверждения "число(A)", указывающие, что значениями основания степени и ее показателя служат числа. Отбрасываются случаи, когда заголовком выражения A служит операция, заведомо принимающая только числовые значения. Далее рассматриваются следующие подслучаи:

1. Знаменатель x6 нечетный; числитель x5 неотрицательный. Если b отлично от 0, то выдается результат x8. Иначе - к x8 предварительно присоединяется утверждение $a \neq 0$.
2. Знаменатель нечетный; числитель отрицательный. К x8 присоедняется утверждение $a \neq 0$ и выдается результат.
3. Знаменатель четный; числитель положительный. К x8 присоединяется утверждение $0 \leq a$ и выдается результат.
4. Знаменатель четный; числитель отрицательный. К x8 присоединяется утверждение $0 < a$ и выдается результат.

Если b не представимо в виде числовой дроби, то переменной x5 присваивается список утверждений из контекста вхождения степенного выражения. Переменной x6 присваивается накопитель результата, заполняемый утверждениями "число(...)", как и выше. Если основание степени равно 0, то предпринимается попытка усмотреть, что показатель b положителен. Для этого используется проверочный оператор "усмменьше". Если попытка оказалась неудачной, в накопитель x6 добавляется утверждение $0 < b$. Если x1 - задача на преобразование, имеющая цель "теоремаприема", то сразу выдается результат x6. Это - особый случай, используемый генератором приемов. Иначе - анализ ситуации продолжается.

Если удается усмотреть, что значением выражения b является рациональное число, то к выражению "знаменатель(b)" применяется пакетный нормализатор "нормзнаменатель". Если удается усмотреть, что это выражение имеет своим значением нечетное число, то проверяется, является ли основание степени десятичной записью числа, отличного от 0. Если является, то выдается результат x6. Если не является, то предпринимается попытка усмотреть положительность показателя степени. При удаче выдается результат x6, иначе - к x6 добавляется утверждение $a \neq 0$, и тоже выдается результат.

В остальных случаях к накопителю x6 будет добавлено утверждение о положительности или неотрицательности основания степени (если оно отсутствовало в контексте). При усмотрении положительности показателя степени вводится условие неотрицательности основания, иначе - положительности. Лишь в двух специальных случаях условие неотрицательности здесь заменяется более сильным условием положительности. Один случай связан с определением о.д.з. во время работы вспомогательной процедуры "сопровождтерм", используемой для инициализации комментариев (сопровождение ...). Другой случай относится к вычислению пределов.

Все выполнявшиеся выше быстрые проверки обеспечивались проверочными операторами элементарной алгебры, реализованными (за исключением нескольких приемов) на ГЕНОЛОГе.

9.2 Процедуры, используемые при идентификации

Приводимые ниже процедуры, используемые при идентификации, вставляются в программы приемов компилятором ГЕНОЛОГа. При необходимости можно отменить их использование, сопроводив прием соответствующими указателями.

9.2.1 Процедура "алгебрпересечение"

Если идентифицируются выражения $f(x, a)$ и $f(x, b)$, где x, a, b - еще не идентифицированные различные переменные, причем f - ассоциативная и коммутативная операция, то компилятор может принять решение определять x как максимальную "общую часть" двух идентифицирующих термов A, B. В простейших случаях (например, для теоретико-множественных операций $\cap, \cup$) эта общая часть представляет собой пересечение множеств корневых f- операндов термов A, B. Однако, при наличии специальных обозначений "степенного" типа для выражений $f(c, c, \ldots, c)$, константных операндов, одноместных операций, которые можно "выносить" из-под операции f, - определение общей части можно понимать более широко. Например, естественно понимать под максимальной общей частью выражений $-2a^2b^3$, $4a^3b^2$ выражение $2a^2b^2$.

Для нахождения такой общей части служит созданная на ЛОСе процедура "алгебрпересечение". Она имеет следующие входные данные: x1,x2 - набор множителей выражений A, B, для которых определяется максимальная общая часть. x3 - набор утверждений, предполагаемых истинными при рассмотрении данных выражений. x4 - 0 либо 1 - указатель режима.

При x4 $= 0$ находится наибольший общий делитель d одночленов A, B (возможно, предварительно преобразованных, чтобы в него вошло возможно больше сомножителей). Выходным переменным x5, x6 присваиваются наборы сомножителей частных от деления A, B на d. Переменной x7 присваивается набор сомножителей выражения d; переменной x8 - список утверждений из x3, использованных для обоснования корректности преобразований, позволивших выделить d.

При x4 $= 1$ вместо нахождения наибольшего общего делителя предпринимается проверка делимости (как и выше, с возможными вспомогательными тождественными преобразованиями) выражения B на A. Если она имеет место, то x5:$=$ символ "пустоеслово"; x6:= набор сомножителей частного от деления B на A; x7:= x1; x8:= список утверждений, использованных для обоснования корректности. Этот режим применяется, если значение переменной a уже идентифицировано с выражением A, и нужно идентифицировать ax с B.

Прежде, чем перейти к рассмотрению программы процедуры "алгебрпересечение", заметим, что компилятор находит ее название с помощью справочника "пересечениесписков" по символу "умножение". Формат входных и выходных данных процедур, определяемых этим справочником, совпадает с приведенным выше форматом процедуры "алгебрпересечение". Фактически этот справочник понадобился пока лишь дважды: в комплекснозначном случае создана аналогичная процедура "Алгебрпересечение".

Выходим на начало программы символа "алгебрпересечение" через пункт ("Приемы решателя", "Элементарная алгебра", "Оператор АЛГЕБРПЕРЕСЕЧЕНИЕ") оглавления программ.

Прежде всего, анализируются знаки выражений. Если выражение A имело знак "минус", то набор x1 его формальных сомножителей - операндов операции "умножение" - будет одноэлементным и состоящим из самого A. В этом случае индикатор знака x9 полагается равным 1, а переменной x1 переприсваивается набор сомножителей выражения, полученного из A отбрасыванием знака "минус". При отсутствии знака "минус" индикатору x9 присваивается 0, а переменной x1 переприсваивается копия исходного набора. Теперь, при любых изменениях разрядов набора x1, используемая внешней процедурой исходная версия x1 будет сохранена. Такие же действия предпринимаются для набора x2, причем роль индикатора знака здесь играет переменная x10.

После контрольной точки "прием(2)" вводится накопитель общих множителей x11 и накопитель x12 использованных утверждений списка x3. Если хотя бы один из наборов x1, x2 имеет степенное выражение, основанием которого служит десятичное число, то каждая натуральная константа, встречающаяся в наборах x1 и x2, представляется в виде произведения степеней простых множителей, и заменяется на набор этих степеней. При этом все общие элементы наборов x1, x2 переносятся в набор x11 и исключаются из x1,x2. Индикатор x13 разложения констант на простые множители полагается равным 1. Если же наборы x1,x2 не имели степеней с численным основанием, этот индикатор остается равным 0. Далее - переход через "ветвь 3" к фрагменту, содержащему контрольную точку "прием(3)".

Здесь начинается цикл просмотра сомножителей, представленных в наборе x1. Единичные сомножители пропускаются. Переменной x15 присваивается текущий сомножитель.

Если индикатор x13 равен 0, то анализируется случай, в котором x15 - натуральная константа. Переменной x16 присваивается ее численное значение, и просматриваются все натуральные константы x18 набора x2. Определяется наибольший общий делитель x20 величин x16, x18. Если он отличен от 1, то в накопитель результата x11 заносится десятичная запись числа x20. При этом x18 и x16 сокращаются на x20. Если в некоторый момент x16 становится равным 1, то просмотр списка x2 обрывается, и переход к очередному элементу списка x1. Если список x2 полностью просмотрен, но x16 не равно 1, то проверяется, равен ли единице указатель режима x4. Если он равен единице, т.е. нужно было проверить делимость B на A, то выход по значению "ложь": делимости нет уже для численных коэффициентов. Иначе - выражение x15 заменяется в списке x1 на десятичную запись величины x16, и продолжение просмотра списка x1.

Если x13 равно 1 либо x15 не является натуральной константой, то переход через "ветвь 2" к фрагменту, содержащему контрольную точку "прием(11)". Здесь анализируется подслучай, когда x15 есть рациональная степень произведения, а все сомножители $p_1, \ldots, p_k$ этого произведения суть элементы списка x2. Проверяется, что числитель m показателя степени меньше знаменателя, а знаменатель n четный (при нечетном знаменателе ранее сработали бы приемы, преобразующие степень произведения в произведение степеней). С помощью оператора "усмменьшеилиравно" проверяется, что основание степени неотрицательно. Этому оператору передается комментарий "алгебрпересечение", блокирующий приемы, которые в свою очередь могли бы обратиться к процедуре "алгебрпересечение". При успешной проверке сомножители $p_1, \ldots, p_k$ заменяются в списке x2 на выражение $(p_1 \ldots p_k)^{(n-m)/n}$; выражение x15 заносится в накопитель x11, а на место его вхождения в список x1 помещается единица. Кроме того, в списке x12 регистрируются утверждения, использованные проверочным оператором. Далее - продолжение просмотра списка x1.

Если x15 не имело указанного степенного вида, то переход через "ветвь 1" к фрагменту, содержащему контрольную точку "прием(5)". Если выражение x15 имеет вид степени, то переменной x16 присваивается вхождение основания этой степени, а переменной x17 - показатель степени. В противном случае выражение x15 рассматривается как вырожденная степень; переменной x16 присваивается вхождение корня этого выражения, а переменной x17 - единица (в формате терма). Затем начинается просмотр выражений списка x2. Переменной x19 присваивается текущее такое выражение; переменной x20 - вхождение основания его степени (как и для x15, допускается случай вырожденной степени).

Сначала разбирается случай совпадения оснований степени выражений x15, x19. Здесь переменной x21 присваивается показатель степени выражения x19. Если показатели x21 и x17 совпали, то выражение x15 заносится в накопитель x11, а его вхождения в списки x1, x2 заменяются на единицы. В этом случае происходит откат к рассмотрению очередного элемента списка x1. Если показатели различны, то переход через "ветвь 4".

Здесь сначала рассматривается случай, когда оба показателя x21, x17 суть рациональные константы. Переменным x22, x23 присваиваются числитель и знаменатель показателя x17, а переменным x24, x25 - числитель и знаменатель показателя x21. Если хотя бы один из знаменателей четный и не усматривается неотрицательность основания степени, то переход к очередному элементу списка x2. Иначе - переход через "ветвь 2" к фрагменту, содержащему контрольную точку "прием(6)".

Находятся числитель x26 и знаменатель x27 результата вычитания показателя x17 из показателя x21. Если x26 меньше нуля, то наибольшим общим делителем выражений x15, x19 является выражение x19. Оно заносится в накопитель x11 и заменяется в списке x2 на единицу. При этом x15 делится на x19, а x17 переприсваивается новый показатель степени выражения x15. Изменения списка x1 здесь пока не происходит - оно выполняется по окончании цикла просмотра списка x2. Если x26 неотрицательно, то наибольшим общим делителем выражений x15, x19 служит выражение x15. Оно заносится в накопитель x11 и заменяется в списке x1 на единицу. Выражение x19 делится в списке x2 на x15, и откат к просмотру очередного элемента списка x1.

Если показатели x21, x17 не являются рациональными константами, то переход через "ветвь 1" к фрагменту, содержащему контрольную точку "прием(10)".

Каждое слагаемое выражений x17, x21 представляется в виде $(m/n)P$, где m, n - числитель и знаменатель численного коэффициента данного слагаемого. Переменной x22 присваивается пара наборов троек (P, m, n): первый набор строится по слагаемым выражения x17, второй - по выражению x21. Инициируется пустым словом накопитель x23 троек аналогичного вида, соответствующий слагаемым показателя степени наибольшего общего делителя множителей x15, x19. Для его заполнения предпринимается просмотр троек x25, соответствующих показателю x17. При фиксированной тройке x25 $= (P, m, n)$ находится тройка x27 показателя x21, имеющая вид (P, q, r). Если дроби $m/n, q/r$ равны, то копия x25 заносится в накопитель x23, после чего первые элементы троек x25, x27 заменяются на нули. В противном случае выбирается меньшая дробь; копия ее тройки переносится в x23, а первый элемент тройки заменяется на 0. Из большей дроби в противоположной тройке вычитается меньшая.

По окончании сравнения слагаемых показателей степени x17, x21 проверяется непустота накопителя x23. Далее, проверяется неотрицательность общего основания x16 рассматриваемых степеней. Переменной x24 присваивается тройка выражений, полученных при возведении x16 в степени - суммы, определяемые, соответственно, накопителем x23 и преобразованными наборами пары x22. Первое из этих выражений представляет собой наибольший общий делитель множителей x15, x19. Оно заносится в накопитель x11. Два других суть результаты деления x15, x19 на выделенный общий множитель. Первый из них присваивается переменной x15; второй - заносится на текущую позицию списка x2 вместо старой версии x19. Если x15 обратилось в единицу, то откат к рассмотрению очередного элемента списка x1, иначе - откат к рассмотрению очередного элемента списка x2.

Вернемся к той точке, где сравнивались основания степени выражений x15, x19. Это происходило в фрагменте с контрольной точкой "прием(5)". Для рассмотрения особых случаев, когда возможно выделение общего множителя у степеней с различными основаниями, переходим через "иначе 3" после оператора "равныетермы(x16 x20)".

Прежде всего, проверяется, не являются ли основания степеней x15, x19 суммами, получающимися друг из друга изменением знаков всех слагаемых (контрольная точка "прием(7)"). Если являются, то проверяется, представим ли показатель степени x17 в виде простой дроби с нечетным знаменателем. После этого x15 преобразуется к тому же основанию степени, что у выражения x19, с возможным изменением индикатора знака x9. Если знаменатель дроби четный, причем x4 $= 1$ (случай проверки делимости), то проверяется, представим ли в виде простой числовой дроби с нечетным знаменателем показатель степени выражения x19. Если представим, то x19 преобразуется к тому же основанию, что у x15, с модификацией индикатора знака x10. После коррекции x15 - откат к повторному просмотру списка x2; после коррекции x19 - откат к повторному рассмотрению текущего элемента списка x2.

Следующий особый случай - одно из оснований степени равно другому, заключенному под знак модуля. Если первая степень (без модуля в основании) имеет рациональный показатель с четным числителем, то ее основание заносится под знак модуля. После этого основания становятся равны, и выполняется откат к повторному рассмотрению ситуации.

По окончании цикла просмотра списка x1 - откат к корневому фрагменту процедуры и переход через оператор "ветвь 2". На данный момент сохраняются значения накопителей x11 (множители, отнесенные к общему делителю) и x12 (утверждения, использованные для обоснования корректности). Переменные начиная с x13 не определены. Выполняется стандартизация списков x1, x2 для создания по ним результирующих выражений. Если в одном и том же списке встречаются два степенных выражения с одинаковыми основаниями и рациональными показателями, то они перемножаются. Если встречается натуральная запись десятичного числа, причем оценка числа разрядов результата возведения в степень меньше 15, то эта степень вычисляется. Если имеется дробная степень натурального числа, у которой числитель дроби больше знаменателя, то выделяется целочисленный множитель, на который домножается другая целочисленная константа списка. Наконец, перемножаются числовые константы одного и того же списка.

После стандартизации списков x1, x2 переменным x13, x14 присваиваются результаты отбрасывания из них всех единичных множителей. Эти переменные суть наборы множителей, полученных после сокращения исходных выражений на общий делитель x11. Если x4 = 1, проверяется, что список x13 пуст. Далее элементы списков лексикографически упорядочиваются; учитываются индикаторы знаков, и выдается результат. Заметим, что при изменении знака формируется новый одноэлементный список, образованный выражением "минус($p_1 \dots p_n$)", где $p_1, \dots, p_n$ - элементы старого списка.

Компилятор ГЕНОЛОГа использует процедуру "алгебрпересечение" без каких-либо специальных дополнительных указаний. Однако, эта процедура сравнительно трудоемкая, и там, где это возможно, лучше обходиться без нее - рассматривая обычные пересечения либо разности групп сомножителей. Чтобы заблокировать использование компилятором данной процедуры, следует ввести указатель приема "набороперандов($A_1 \dots A_n$)", где $A_1, \dots, A_n$ - указатели вхождений в теорему приема тех произведений, которые должны обрабатываются без ее участия.

9.2.2 Процедура "выделениестепени"

Для идентификации степенных выражений вида a^n, где n - натуральная константа, используется процедура "выделениестепени(x1 x2 x3 x4 x5 x6)". Здесь x1 - вхождение первого символа выражения, идентифицируемого со степенью; x2 - натуральный показатель степени в формате числа ЛОСа; x3 - список утверждений, истинных в контексте вхождения x1; x4 - список комментариев. Используя указания, содержащиеся в комментариях списка x4, процедура принимает решение о целесообразности представления идентифицируемого выражения в виде n-й степени некоторого другого выражения. Если это целесообразно, переменной x5 присваивается основание результирующей степени, а переменной x6 - список утверждений, использованных для обоснования корректности.

При принятии решения о целесообразности выделения степени может анализироваться внешний контекст: если уже встречались выражения, возникающие при выделении степени, либо корни из таких выражений, то решение принимается. Внешний контекст определяется с учетом следующих комментариев списка x4:

1. "выделениестепени" - внешний контекст не рассматривается;
2. "нормуравнения", "базавхождения" - внешний контекст образован тем термом, который обрабатывается внешним пакетным нормализатором;
3. "(степень A)" - к внешнему контексту добавляется терм A.
4. "уравнменьше"; "следлиния" - из внешнего контекста исключаются все степени, содержащие неизвестные;

По умолчанию, внешним контекстом являются условия текущей задачи. Степенные выражения, входящие во внешний контекст, перечисляются вспомогательной процедурой "степени(...)", которой передаются входные параметры x1, x4.

Переходим к рассмотрению программы процедуры. Выйти на нее можно через пункт ("Приемы решателя", "Элементарная алгебра", "Оператор ВЫДЕЛЕНИЕСТЕПЕНИ") оглавления программ.

Прежде всего, проверяется, равно ли x1 нулю. Если равно, то сразу же выдается результат: x5 - однобуквенный терм "0", x6 - "пустоеслово". В противном случае переменной x7 присваивается набор сомножителей выражения x1. Вводятся накопитель x8 сомножителей результата и накопитель x9 использованных утверждений.

После контрольной точки "прием(2)" начинается цикл обработки константных сомножителей. Здесь вводятся еще два накопителя: x10 - простые основания степеней; x11 - пары (числитель - знаменатель) для соответствующих показателей. Просматриваются такие элементы списка x7, которые представляют собой либо десятичные записи натуральных чисел, либо степени натуральных чисел с рациональным константным показателем. Для представления оснований степени этих элементов в виде произведений натуральных степеней простых чисел используется процедура "множители(...)". Результат рассмотрения очередного элемента регистрируется в списках x10, x11, а вхождение элемента в список x7 заменяется на 0.

По окончании просмотра списка x7 выполняется коррекция элементов списка x11, соответствующая извлечению из константных множителей корня x2-й степени. Для этого списки x10, x11 просматриваются синхронно. Пусть p - текущий элемент списка x10; (m, n) - текущий элемент списка x11. Если m делится на x2, то такое деление выполняется.

Иначе - проверяется наличие комментария "выделениестепени". При его отсутствии последовательно просматриваются степенные термы t внешнего контекста, представляющие собой рациональные степени натуральных констант a, делящихся на p. Если результат деления дроби m/n на рациональный показатель, с которым простое число p входит в t, есть натуральное число, делящееся на x2, то выполняется домножение n на x2 и сокращение дроби m/n. Затем - переход к очередным элементам списков x10, x11. При нарушении этих условий - выход из процедуры по значению "ложь".

По окончании просмотра списков x10, x11 реализуется цикл заполнения накопителя результата x8, куда заносятся степенные выражения, определяемые элементами p, (m, n) списков x10, x11. Из этих выражений, по мере возможности, извлекаются натуральные константы, и вычисляется произведение x12 всех таких констант. Если в результате число x12 оказывается отлично от единицы, то его десятичная запись добавляется к списку x8.

По завершении рассмотрения константных множителей - откат к корневому фрагменту и переход через "ветвь 3" к фрагменту, содержащему контрольную точку "прием(5)". Здесь происходит повторный просмотр списка x7 - на этот раз уже не только константных, а произвольных его элементов. Нулевые значения, появившиеся после первого цикла просмотра, пропускаются. Если текущий элемент x10 - степенное выражение с рациональным показателем степени, числитель которого делится на x2, то из x10 сразу же извлекается корень x2-й степени, и результат заносится в накопитель x8. Это происходит после контрольной точки "прием(6)". Если x10 не имеет указанного вида, то переход через "ветвь 2".

Здесь переменной x11 присваивается основание степени (быть может, вырожденной) выражения x10. Если выражение x10 - степенное, имеет рациональный показатель степени p, а в наборе x4 нет элемента "выделениестепени", то просматриваются степенные термы x15 внешнего контекста, основание которых равно x11. Если x15 имеет рациональный показатель q, причем отношение p/q - целое число, делящееся на x2, то корень x2-й степени из x10 заносится в x8. В случае четного x2 проверяется неотрицательность x11.

Если выражение x10 - степенное, а показатель его не является рациональной константой, то последовательно просматриваются слагаемые s этого показателя. Заполняется накопитель x13 результатов R деления таких слагаемых на x2. Если выделяется натуральный множитель числителя слагаемого, делящийся на x2, то R сразу заносится в x13. Иначе - при отсутствии комментария "выделениестепени" просматриваются степенные термы внешнего контекста, основание которых совпадает с основанием x10, а некоторое слагаемое r показателя отличается от s только рациональным коэффициентом. Коэффициент слагаемого s делится на коэффициент слагаемого r; если получается целочисленный результат, делящийся на x2, то R заносится в накопитель x13. Если для s ни одно из указанных условий не выполнено, то выход из процедуры по значению "ложь". Иначе, по завершении цикла просмотра слагаемых, в накопитель x8 заносится степенное выражение с основанием x11 и показателем, равным сумме выражений набора x13. Затем - переход к очередному сомножителю x10.

Наконец, если при просмотре списка x7 встречается не степенное выражение x10, то проверяется отсутствие комментария "выделениестепени" и просматриваются степенные термы внешнего контекста, основанием которых служит x10. Если встречается терм с рациональным показателем, знаменатель которого делится на x2, то в x8 заносится результат извлечения из x10 корня x2-й степени; для четного x2 проверяется неотрицательность основания.

Непосредственно в процедуру вставлен эвристический фильтр, блокирующий выделение степени из неизвестной, для которой в задаче на описание уже найдено ее значение.

По завершении цикла просмотра списка x7 выдается результат - произведение выражений накопителя x8. Это произведение обрабатывается оператором "стандупорядочение", располагающим операнды коммутативных операций и симметричных отношений в стандартном порядке.

Процедура "выделениестепени" используется компилятором ГЕНОЛОГа для идентификации натуральных константных степеней без каких-либо дополнительных указаний. Если нужно заблокировать ее применение, прием сопровождается указателем "вид($A_1 \dots A_n$)", где $A_1, \dots, A_n$ - указатели вхождений в теорему приема соответствующих степенных выражений.

9.2.3 Идентификация многочленов

Иногда бывает необходимо усмотреть, что заданное выражение представляет собой значение некоторого многочлена $P(x)$ в точке A. Эта точка может быть определена заранее либо найдена в процессе идентификации. Различаются два случая. В простейшем случае результатом идентификации служит набор коэффициентов многочлена, которые используются приемом для формирования новых термов. Явного выражения для самого многочлена $P(x)$ здесь получать не требуется, так что набор коэффициентов остается лишь в технических структурах данных программы приема. Такой случай обслуживается процедурой "сммногочлен". Другой случай - получение формальной записи "многочлен(...)" для многочлена $P(x)$, которая далее будет использоваться приемами, ориентированными на работу с многочленами. Тогда применяется процедура "формногочлен".

Начнем с рассмотрения первой процедуры. Обращение к ней имеет вид "сммногочлен(x1 x2 x3 x4)". Здесь x1 - вхождение первого символа идентифицирующего выражения, x2 - указатель на принцип определения выражения A. Если x2 - набор переменных, то A группируется из всех множителей одночленов, содержащих переменные списка x2. Если x2 - символ "0", то A группируется из всех неконстантных множителей одночленов. Наконец, если x2 - терм "терм(A)", то имеем явное задание выражения A. Если удалось усмотреть многочлен, то выходной переменной x3 присваивается выражение A, а переменной x4 - набор коэффициентов многочлена в формате термов, упорядоченных по возрастанию степеней. В этом наборе вставлены нулевые термы для отсутствующих промежуточных степеней. В случае, когда A первоначально не задано, требуется, чтобы x1 имело вид суммы. Если A первоначально было задано, оно не пересылается переменной x3, которая получает значение 0.

Переходим к корневому фрагменту программы символа "сммногочлен". Если x1 не имеет вида суммы, причем A не задано, то сразу выдается отказ. Иначе вводятся накопитель x5 выражения A и накопитель x6 списка пар (натуральный показатель степени выражения A - коэффициент при этой степени), соответствующих слагаемым выражения x1. В этом списке могут появляться несколько различных пар для одного и того же показателя степени. Переменной x7 присваивается список сомножителей выражения A, если оно задано, и 0 в противном случае. Затем начинается цикл просмотра слагаемых выражения x1. Если обнаруживается дробное слагаемое, знаменатель которого имеет общие переменные с x2 (кроме случая x2 = 0), то выдается отказ. Иначе - переменной x11 присваивается список сомножителей числителя текущего слагаемого; знак слагаемого отбрасывается. Если в x11 встречается не натуральная степень неконстантного выражения, являющегося также основанием степени выражения списка x7, то выдается отказ.

В случае, когда выражение A уже задано, находится максимальная его степень x12, на которую делится произведение x11. Здесь используется оператор "алгебрпересечение". В накопитель x6 заносится пара (x12, x13), где x13 - частное от деления, скорректированное с учетом внешних знака и знаменателя.

Если A не задано, то определяется список x12 всех выражений набора x11, имеющих переменную из x2 (при x2 = 0 - всех неконстантных выражений). Если x12 пусто, в накопитель x6 заносится пара (0 - текущее слагаемое). Иначе - создается список x13 пар (основание степени - показатель степени) для выражений списка x12. Переменной x14 присваивается наибольший общий делитель целочисленных множителей показателей, и формируется выражение x16 - результат извлечения из произведения x12 корня x14-й степени. Это выражение стандартизируется и сравнивается с содержимым накопителя x5. Если x5 не равно 0 и отлично от x16, выдается отказ. Если переменная x5 равна 0, то ей присваивается значение x16. Затем в накопитель x6 заносится пара (x14, x17), где x17 - скорректированное с учетом внешних знака и знаменателя произведение элементов списка x11, не вошедших в список x12.

По окончании цикла просмотра слагаемых - переход через "иначе 2". Здесь находится максимальный показатель степени x9 и вводится накопитель x10 набора коэффициентов многочлена, изначально заполненный нулевыми термами. В цикле просмотра списка x6 происходит группировка по разрядам накопителя x10 коэффициентов соответствующих степеней, затем выдается результат.

Процедура "формногочлен(x1 x2 x3 x4)" получает следующие входные данные: x1 - выражение, идентифицируемое как многочлен с константными коэффициентами; x2 - терм либо набор термов, логических символов и переменных, которые должны быть идентифицированы со значениями переменных многочлена; x3

- список утверждений, в предположении истинности которых проводится идентификация. При успехе переменной x4 присваивается выражение "многочлен(X K F)", определяющее идентифицированный формальный многочлен. Напомним, что X - набор формальных переменных многочлена; K - кольцо, над которым рассматривается многочлен; F - функция, задающая коэффициенты. Она определена на множестве наборов целых неотрицательных степеней переменных списка X и принимает значения из K. Обычно такая функция задается с помощью выражения "таблица(перечень(набор($F_1 \dots F_n$)))", где F_i - одноточечные функции "См(a_i b_i)". Выбор переменных X здесь произволен. В выражении для многочлена используются константные выражения "икс(m)", задающие данные переменные, так что сами они не являются параметрами выражения, определяющего многочлена. Заметим, что возможности рассматриваемой процедуры ограничены тремя случаями кольца K - кольцом целых чисел, а также полями вещественных и комплексных чисел.

Программа процедуры вводит накопитель x5 выражений "См(a_i b_i)", задающих коэффициенты многочлена. Переменной x6 присваивается набор слагаемых выражения x1. Переменной x8 присваивается набор значений переменных многочлена, приведенных к формату термов. Далее просматриваются одночлены x9 набора x6. Для текущего одночлена создается набор x10 целых неотрицательных степеней, с которыми в нем встречаются выражения списка x8. Проверяется, что оставшиеся множители одночлена не содержат переменных, находится их произведение x16, и в накопитель x5 заносится выражение "См(x10 x16)". По окончании просмотра одночленов формируется выражение x9 вида "таблица(перечень($F_1 \dots F_n$))" для функции F. Переменной x10 присваивается тип кольца. Комплексный случай усматривается по комплекснозначным операциям "Плюс", "Минус", "Умножение", "Степень". Если в вещественнозначном случае все коэффициенты суть целые числа, то берется кольцо целых чисел, иначе - поле вещественных. В качестве X выбираются переменные, не встречающиеся в утверждениях x3 и выражении x1. Затем выдается результирующее выражение "многочлен(X K F)".

Чтобы компилятор ГЕНОЛОГа воспользовался процедурой "сммногочлен" для идентификации теоремного терма $f(A)$ как значения многочлена f от выражения A, нужно ввести указатель приема "сммногочлен(f R)". Терм R определяет способ идентификации выражения A: если R - логический символ "неизвестные", то A группируется из всех содержащих неизвестные текущей задачи множителей одночленов; если R - символ переменной, то A группируется из всех содержащих идентифицированную с R переменную множителей; если R имеет вид "терм(B)", то A есть значение программного выражения B. Наконец, R вообще может отсутствовать, и тогда A группируется из всех неконстантных множителей.

Процедура "сммногочлен" применяется компилятором также для идентификации многочлена, заданного конечной суммой вида:

$$\sum_{i=1}^{n} f(i)x^{i-1}.$$

В этом случае вводится указатель приема "видмногочлена(U)", где U - указатель вхождения данной конечной суммы в теорему. Предполагается, что переменная x заблаговременно идентифицирована с некоторым выражением A. Переменная f идентифицируется с термом "набор(...)", перечисляющим коэффициенты многочлена, упорядоченные по возрастанию степеней, а переменная n - с увеличенной на единицу степенью многочлена.

Процедура "формmногочлен" применяется компилятором ГЕНОЛОГа для идентификации выражений "значениемн($f\ A$)". В отличие от предыдущих случаев, здесь уже f - не функциональная переменная, а обычная переменная, которую нужно идентифицировать с формальным многочленом. Прием сопровождается указателем "значениемн(f)". A - переменная либо набор переменных, значения которых идентифицируются заблаговременно.

9.3 Вычисления с константами

Начнем с напоминания общих принципов вычислений на ЛОСе, которые приводились в первом томе. Для представления вещественных чисел используются следующие форматы:

1. Символьные числа - логические символы, кодирующие целые числа от -259 до 65275. Символ с номером $260 + n$ кодирует число n. В частности, логический символ "0" кодирует число 0. Этот формат обеспечивает наиболее быструю арифметику. Он применяется для указания номеров разрядов наборов, смещений пикселей на экране, уровней сканирования и т.п.

2. Десятичные числа ЛОСа - логические символы для цифр от 0 до 9, а также наборы цифр, быть может, дополненные логическим символом "минус" в начале и логическим символом ","в середине. Избыточные нули в начале и конце набора, а также набор "минус 0" не допускаются.

3. Представление десятичного числа в виде терма. Цифры представляются однобуквенными термами. Более чем одноразрядное положительное число $a_1 \dots a_n$ представляется термом "величина($a_1 \dots a_n$)". Здесь a_i - цифры, быть может, с единственным вхождением символа ",". Для получения отрицательной константы используется внешняя операция "минус". Термы, представляющие десятичные числа, называем десятичными записями.

4. Для усиления вычислительных возможностей системы введены дополнительные типы "атомарных" объектов ЛОСа - ряд специальных представлений числовых констант:

   (a) Формат "целое со знаком". Он представляет собой специальным образом закодированное обычное 32-битное целое число со знаком.

   (b) Формат "длинное целое со знаком" позволяет работать с целыми числами произвольной длины. Отличие от формата "десятичное число ЛОСа" состоит в более эффективном использовании машинной арифметики: роль цифр здесь играют 16-битные фрагменты.

   (c) Формат "число с плавающей запятой" - специальным образом закодированное 64-битное стандартное представление числа "с плавающей запятой".

В обычном "логическом" режиме работы решателя данные форматы не используются. Для их подключения нужны специальные целевые установки задач, указывающие на смешанный "логико-вычислительный" режим. Действия с логическими структурами данных, предполагающие режим сканирования, обычно на порядок медленнее, чем обычные вычисления. Поэтому для смешанного режима нет необходимости в дальнейшем ускорении вычислительного потенциала системы. Чтобы обеспечить такое ускорение для чисто вычислительного режима, например, в задачах моделирования больших систем, нужно развивать технику синтеза решателями "обычных" СИ-программ, которые могли бы запускаться самим же решателем с целью получения им необходимых данных.

Напомним основные операторы ЛОСа, используемые для работы с числами.

9.3.1 Символьные числа

1. "менее(x1 x2)" - проверка того, что число x1 меньше числа x2;
2. "плюссимв(x1 x2)" - сумма чисел x1, x2;
3. "вычитсимв(x1 x2)" - разность чисел x1 и x2;
4. "умножсимв(x1 x2)" - произведение чисел x1 и x2;
5. "частноесимв(x1 x2 x3 x4)" - деление с остатком x1 на x2 (x3 - неполное частное, x4 - остаток);
6. "симвномера(x1 x2 x3)" - перечисление символьных чисел x3 в порядке возрастания начиная с символьного числа x1 и кончая x2;
7. "обрномера(x1 x2 x3)" - перечисление символьных чисел x3 в порядке убывания начиная с x1 и кончая x2.

9.3.2 Десятичные числа ЛОСа; общий случай

.

1. "меньше(x1 x2)" - проверка того, что число x1 меньше числа x2;
2. "больше(x1 x2)" - проверка того, что число x1 больше числа x2. Предпочтительнее проверка через "меньше" - она выполняется быстрее.
3. "плюс(x1 x2)" - сумма чисел x1 и x2;
4. "вычитание(x1 x2)" - вычитание чисел x1 и x2;
5. "умножение(x1 x2)" - произведение чисел x1 и x2;
6. "минус(x1)" - изменение знака числа x1;
7. "модуль(x1)" - абсолютная величина числа;
8. "возведениевстепень(x1 x2)" - результат возведения числа x1 в целую неотрицательную степень x2;
9. "минимум(x1 x2)" - меньшее из чисел x1, x2;
10. "максимум(x1 x2)" - большее из чисел x1, x2;
11. "целаячасть(x1)" - целая часть числа x1;

12. "нормализация(x1 x2)" - результат приведения к стандартному виду числа набора x2 цифр, быть может, имеющего в середине запятую и минус в начале. x1 - указатель необходимости отбрасывания нулей в конце числа (1 - отбрасываются, 0 - нет). Такое отбрасывание нужно для наборов с запятой. Отбрасываются лишние нули в начале набора; если остается единственная цифра, то одноэлементный набор заменяется на нее.
13. "неполноечастное(x1 x2)" - целая часть от деления неотрицательного числа x1 на положительное число x2. Числа не обязательно целые.

9.3.3 Целые десятичные числа ЛОСа

1. "деление(x1 x2 x3 x4)" - деление с остатком целого числа x1 на натуральное число x2. x3 - неполное частное, x4 - неотрицательный остаток.
2. "целое(x1)" - проверка того, что число x1 целое;
3. "натуральное(x1)" - проверка того, что число x1 натуральное;
4. "четное(x1)" - проверка того, что число x1 целое и четное;
5. "множители(x1 x2 x3)" - натуральное число x1 разлагается в произведение степеней простых чисел. x2 становится равно набору этих простых чисел, x3 - набору натуральных показателей их степеней.
6. "нок(x1 x2)" - наименьшее общее кратное натуральных чисел x1 и x2;
7. "нод(x1 x2)" - наибольший общий делитель целых чисел x1, x2;
8. "делители(x1 x2)" - x2 перечисляет все целочисленные делители целого числа x1 (если x1 равно 0, оператор ложен);
9. "простыеделители(x1 x2)" - x2 перечисляет все простые делители натурального числа x1;
10. "делитель(x1 x2 x3)" - x1 есть набор простых чисел; x2 - набор показателей их степеней. x3 перечисляет все натуральные делители произведения данных степеней.
11. "числоделителей(x1)" - число делителей целого числа x1;
12. "частное(x1 x2 x3)" - если целое число x1 делится на целое число x2, то x3 становится равно частному, иначе - оператор ложен;
13. "извлечениекорня(x1 x2 x3)" - если из целого числа x1 извлекается корень натуральной степени x2, то x3 перечисляет результаты извлечения корня (один либо два отличающихся знаком);
14. "степеньделителя(x1 x2 x3 x4)" - x1 и x2 суть целые числа. Если x2 не равно плюс-минус единице, то определяется наибольшее натуральное x3, такое, что x1 делится на x3-ю степень числа x2. При этом x4 становится равно частному от деления x1 на данную степень.
15. "простое(x1)" - x1 перечисляет простые числа;

16. "Разбиения(x1 x2 x3)" - x1 и x2 суть натуральные числа. x3 перечисляет все разбиения числа x1 (наборы натуральных чисел, сумма которых равна x1), начинающиеся с элемента, не превосходящего x2.

17. "делениепомодулю(x1 x2 x3 x4)" - x1 и x2 суть целые числа от 0 до (x3 -1), x3 - натуральное. Если возможно деление x1 на x2 по модулю числа x3, то x4:= результат деления.

18. "Номера(x1 x2 x3)" - перечисление всех целых чисел x3 начиная с целого числа x1 и до целого числа x2;

19. "Целое(x1)" - перечисление целых чисел x1 в порядке возрастания их абсолютной величины $(0, 1, -1, 2, -2, \ldots)$.

9.3.4 Вычисления с десятичными числами ЛОСа, обеспечивающие получение результата с недостатком либо с избытком

Ряд операторов выполняет вычисления с сохранением заданного числа знаков после запятой и округлениями, обеспечивающими получение результата с избытком либо с недостатком. Они используют входной параметр - указатель округления, уточняющий, какой именно случай рассматривается. Если этот указатель равен логическому символу "меньше", то результат получается с недостатком; если он равен логическому символу "не", то результат получается с избытком. Входной параметр, равный числу знаков после запятой, не гарантирует получения заданного числа точных знаков - он лишь ограничивает длину используемых десятичных записей. Обычно применяется объединенный оператор "числоценка", из которого происходят обращения к другим операторам указанного типа, ориентированным на различные классы элементарных функций. Некоторые из процедур используют заблаговременно вычисленные с большим числом знаков константы (e, π и др.). При необходимости можно заменить эти константы на другие, имеющие большее число знаков. Во всяком случае, степень точности данных констант не влияет на гарантию получения результата с избытком либо с недостатком.

1. "округление(x1 x2 x3)" - x1 есть десятичное число ЛОСа; x2 - число знаков после запятой; x3 - указатель округления (символы "не", "меньше", а символ "0", указывающий на округление к ближайшему значению). Значением операторного выражения служит результат указанного округления. В приводимых ниже операторах указатель округления может принимать только значения "не", "меньше".

2. "десделение(x1 x2 x3 x4 x5)" - x1 и x2 суть десятичные числа; x2 не равно 0; x3 - целое неотрицательное число. Определяется результат x4 деления x1 на x2 с точностью до первых x3 знаков после запятой и с недостатком в смысле модуля. Если результат деления точный, x5 становится равно 0, иначе - 1.

3. "десчастное(x1 x2 x3 x4)" - x1 и x2 суть десятичные делимое и делитель; x3 - число знаков после запятой; x4 - указатель округления. Значением операторного выражения служит результат деления x1 на x2, округленный с учетом x4 и дополненный при необходимости нулями в конце мантиссы так, чтобы общее число знаков после запятой равнялось x3.

4. "квадркорень(x1 x2 x3)" - x1 есть десятичное число; x2 - указатель округления; x3 - число знаков после запятой. Значением выражения служит результат извлечения из x1 квадратного корня.

5. "натурлог(x1 x2 x3)" - x1 есть десятичное число; x2 - число знаков после запятой; x3 - указатель округления. Значением выражения служит результат вычисления натурального логарифма числа x1.

6. "эксп(x1 x2 x3)" - x1 есть десятичное число; x2 - указатель округления; x3 - число знаков после запятой. Значением выражения служит результат вычисления экспоненты числа x1.

7. "тригоценка(x1 x2 x3 x4 x5)" - x1 есть десятичное число; x2 - логический символ, являющийся названием тригонометрической функции (синус, косинус, тангенс, котангенс, арксинус, арккосинус, арктангенс); x3 - число знаков после запятой; x4 - указатель округления. x5 становится равно результату вычисления значения функции x2 от числа x1.

8. "числоценка(x1 x2 x3 x4)" - x1 есть вхождение первого символа константного подвыражения; x2 - число знаков после запятой; x3 - указатель округления. x4 становится равно результату вычисления значения выражения x1. Кроме элементарных функций, вычисляемых перечисленными выше операторами, допускается вхождение в x1 констант e,π, а также функций "гипсинус", "гипкосинус" и операции "модуль". Отсутствие в приведенном списке арккотангенса объясняется тем, что он обычно сразу выражается приемом решателя через арктангенс.

9.3.5 Термы, представляющие десятичные константы

При сканировании задач решатель встречает числа, представленные в формате термов. Для преобразований их обычно приходится переводить сначала в формат десятичных чисел ЛОСа. Это делает операторное выражение "числзначение(x1)". Входное данное x1 - вхождение в некоторый терм десятичной записи. Чтобы убедиться, что x1 - действительно вхождение указанного типа, помогает оператор "десчисло(x1)". После того, как вычисления в том или ином вычислительном формате проделаны, для регистрации результата в задаче нужно вернуться к формату десятичных записей. Эту работу выполняет операторное выражение "десзапись(x1)", получающее на вход десятичное число ЛОСа x1. Его значением является соответствующий терм. Кроме перечисленных трех основных операторов ("десчисло", "числзначение", "десзапись"), создан ряд вспомогательных операторов. Обычно они получают входные данные в формате десятичных записей и выдают результат в том же формате:

1. "сложить(x1)" - терм x1 имеет вид суммы двух десятичных записей и преобразуется к виду десятичной записи суммы. Чтобы воспользоваться операторным выражением такого типа, его можно ввести как нормализатор в приеме ГЕНОЛОГа.

2. "сложитьдроби(x1)" - терм x1 имеет вид суммы двух слагаемых, каждое из которых - либо десятичная запись, либо (с точностью до знака) отношение десятичных записей натуральных чисел. Значением операторного выражения служит результат преобразования x1 к виду терма - десятичного числа либо простой несократимой дроби.

3. "умножить(x1)" - терм x1 имеет вид произведения двух десятичных записей. Он преобразуется к виду десятичной записи.

4. "натурстепень(x1)" - терм x1 имеет вид степени, в основании которой находится десятичная запись, а в показателе - десятичная запись целого неотрицательного числа. Терм преобразуется к виду десятичной записи.

5. "смешаннаядробь(x1 x2 x3 x4)" - тройка (x2,x3,x4) определяет координату вхождения в задачу x1 некоторого выражения. Проверяется, что вхождение корневое, а выражение имеет вид смешанной числовой дроби - суммы целого числа и простой несократимой дроби, у которой модуль числителя меньше знаменателя, причем знаки слагаемых одинаковы.

6. "деспорядок(x1 x2 x3)" - x1 есть вхождение десятичной записи положительного числа. x2 становится равно десятичной записи натурального числа, полученного из x1 умножением на наименьшую неотрицательную степень десяти. x3 становится равно показателю этой степени в формате числа ЛОСа.

7. "простаядробь(x1 x2 x3)" - x1 есть вхождение первого символа выражения. Проверяется, что данное выражение есть либо десятичная запись, либо отношение десятичных записей натуральных чисел, быть может, с внешним знаком "минус". Определяется представление x1 в виде несократимой простой дроби. x2 становится равно ее числителю, а x3 - знаменателю. Оба они представлены в формате чисел ЛОСа. Знак минус, если он есть, передается числителю.

8. "констцелое(x1)" - истинно, если x1 есть терм либо вхождение первого символа терма, являющегося десятичной записью целого числа;

9. "констдробь(x1)" - истинно, если x1 есть дробь с натуральными числителем и знаменателем либо получено из такой дроби добавлением внешнего минуса;

10. "дробнаязапись(x1)" - x1 есть десятичная запись с запятой. Значением операторного выражения является терм - результат преобразования x1 к виду простой несократимой дроби, быть может, взятой со знаком минус;

11. "составнаядробь(x1)" - x1 есть отношение десятичных записей натуральных чисел, у которого числитель не меньше знаменателя. Значением операторного выражения является результат преобразования x1 к виду суммы натурального числа и простой несократимой дроби, числитель которой меньше знаменателя.

12. "дробнаявеличина(x1)" - истинно, если x1 есть вхождение первого символа десятичной записи либо числовой дроби (быть может, имеющей внешний знак "минус");

13. "десдробь(x1 x2)" - x1 и x2 суть целое и натуральное числа. Осуществляется сокращение дроби с числителем x1 и знаменателем x2; если знаменатель результата есть степень десяти, то выполняется переход к десятичной записи. Иначе - значением операторного выражения служит терм для найденной несократимой дроби.

14. "сдвигзапятой(x1 x2)" - x1 есть вхождение десятичной записи, x2 - натуральное число ЛОСа. Предпринимается сдвиг запятой в x1 на x2 разрядов влево. Результат представлен в формате десятичной записи.

15. "стандстепень(x1 x2 x3 x4)" x1 есть терм, имеющий вид дробной степени натурального числа. x2, x3 - числитель и знаменатель показателя этой степени; оба в формате чисел ЛОСа. Выполняется переход к стандартному представлению x1: усматриваются сомножители основания, допускающие извлечение соответствующего корня, которые выносятся наружу. x4 становится равно набору термов - сомножителей результата стандартизации.
16. "сокращдроби(x1)" - x1 есть терм, имеющий вид отношения двух десятичных записей. Этот терм преобразуется к виду простой несократимой дроби, возможно, с внешним знаком "минус".
17. "деснабор(x1)" - x1 есть набор десятичных чисел ЛОСа. Значением операторного выражения служит терм, значением которого является набор x1. Это и следующее выражения используются компилятором ГЕНОЛОГа для запаковки-распаковки числовых наборов в приемах, часть антецедентов которых обрабатываются вычислительными процедурами.
18. "числнабор(x1)" - x1 есть выражение "набор($A_1 \dots A_n$)", где $A_1, \dots, A_n$ - десятичные записи. Операторное выражение переходит к набору соответствующих чисел ЛОСа.

9.3.6 Основные операторные выражения для работы с числами, представленными в форматах "целое со знаком", "длинное целое со знаком", "число с плавающей запятой"

Эти выражения имеют вид "выч(...)" и реализуются интерпретатором ЛОСа:

1. "выч(целое x1)", "выч(Целое x1)", "выч(число x1)" - x1 есть десятичное число ЛОСа. Оно переводится, соответственно, в форматы "целое со знаком", "длинное целое", "с плавающей запятой".
2. "выч(выход x1)" - x1 есть представление числа в одном из форматов "целое со знаком", "длинное целое", "с плавающей запятой". В первых двух случаях x1 переводится в формат десятичного числа ЛОСа; в последнем - преобразуется в пару чисел ЛОСа (A_1, A_2), где x1 равно произведению A_1 на 10 в степени A_2. По возможности, A_2 равно 0.
3. "выч(значение x1 x2)" - x1 есть ссылка на массив чисел в формате "целое со знаком" либо "с плавающей запятой"; x2 - номер элемента массива, представленный в формате "целое со знаком". Значением операторного выражения служит данный элемент массива; формат его не изменяется.
4. "выч(плюс x1 x2)", "выч(минус x1)", "выч(вычитание x1 x2)", "выч(умножение x1 x2)" - арифметические операции для чисел в любом из трех рассматриваемых форматов.
5. "выч(дробь x1 x2)" - результат деления; в случае целых чисел берется неполное частное.
6. "выч(степень x1 x2)" - результат возведения числа x1 в степень x2. Для целых чисел величина x2 должна быть неотрицательной.

7. "выч(логарифм x1 x2)" - x1, x2 представлены в формате "с плавающей запятой". x1 - основание логарифма, x2 - число под логарифмом.
8. "выч(натурлог x1)", "выч(синус x1)", "выч(косинус x1)", "выч(тангенс x1)", "выч(котангенс x1)", "выч(арксинус x1)", "выч(арккосинус x1)", "выч(арктангенс x1)", "выч(арккотангенс x1)", "выч(квадркорень x1)", "выч(эксп x1)", "выч(гипсинус x1)". "выч(гипкосинус x1)", "выч(гиптангенс x1)", "выч(гипкотангенс x1)" - вычисление элементарных функций в формате "с плавающей запятой".
9. "выч(модуль x1)" - вычисление абсолютной величины числа x1, имеющего формат "целое со знаком" либо "с плавающей запятой";
10. "выч(копия x1)" - копия числа, представленного в одном из трех рассматриваемых форматов.
11. "выч(плюсбеск)", "выч(минусбеск)" - константы $2e99$, $-2e99$.
12. "выч(Плюс x1 x2 x3)" - сумма целых неотрицательных чисел x1, x2 по натуральному модулю x3. Здесь и далее в модулярных операциях числа представлены в формате "длинное целое".
13. "выч(Минус x1 x2)" - изменение знака целого неотрицательного числа x1 по модулю x2;
14. "выч(Умножение x1 x2 x3)" - произведение целых неотрицательных чисел x1, x2 по натуральному модулю x3;
15. "выч(Дробь x1 x2 x3)" - частное от деления целого неотрицательного числа x1 на натуральное число x2 по простому модулю x3;
16. "выч(Целые x1)" - результат перевода числа x1 из формата "целое со знаком" в формат "длинное целое со знаком";
17. "выч(нод x1 x2)" - вычисление наибольшего общего делителя для формата "длинное целое";
18. "выч(вычет x1 x2)" - остаток от деления целого числа x1 на натуральное число x2. Формат - "длинное целое".
19. "выч(целаячасть x1)" - целая часть числа x1, представленного в формате "с плавающей запятой".
20. "выч(сигнум x1)" - сигнум числа x1, имеющего формат "с плавающей запятой", переведенный в формат "целое со знаком";
21. "выч(логсимвол x1)" - результат преобразования числа x1 из формата "целое со знаком" в формат символьных чисел;
22. "выч(номерсимв x1)" - результат преобразования символьного числа x1 в формат "целое со знаком";
23. "выч(симвномер x1)" - результат преобразования символьного числа x1 в формат "с плавающей запятой";

9.3.7 Основные операторы для работы с числами, представленными в форматах "целое со знаком", "длинное целое со знаком", "число с плавающей запятой"

Все эти операторы имеют вид "Выч(...)" и реализуются интерпретатором ЛОСа:

1. "Выч(набор x1 x2 x3)" - выделение памяти для массива чисел. x1 - логический символ "число" либо "целое", определяющий формат элементов массива ("с плавающей запятой" либо "целое со знаком"). x2 - натуральное число в формате десятичных чисел ЛОСа, определяющее длину массива. x3 становится равно ссылке на новый массив.

2. "Выч(запись x1 x2 x3)" - x1 есть ссылка на числовой массив; x2 - номер элемента массива; x3 - число, представленное в формате элементов массива x1. Происходит регистрация значения x3 в качестве x2-го элемента массива. Нумерация элементов массива начинается с 0; x2 представлено в формате "целое со знаком".

3. "Выч(деление x1 x2 x3 x4)" - x1 и x2 суть числа в формате "цело со знаком" либо "длинное целое". Выполняется деление с остатком x1 на x2; x3 становится равно неполному частному, а x4 - остатку от деления.

4. "Выч(меньше x1 x2)" - x1 и x2 суть числа в одинаковом формате (одном из трех рассматриваемых). Оператор истинен, если x1 меньше, чем x2;

5. "Выч(меньшеилиравно x1 x2)" - аналогично предыдущему, но проверяется нестрогое неравенство;

6. "Выч(0 x1)" - x1 есть число в одном из трех рассматриваемых форматов. Оператор истинен, если это число не равно 0;

7. "Выч(равно x1 x2)" - x1 и x2 суть числа в одинаковом формате. Оператор истинен, если эти числа равны.

8. "Выч(изменение x1 x2)" - x1 и x2 суть числа, представленные в формате "целое со знаком" либо "с плавающей запятой". Значение x2 переносится в x1. Заметим, что значением переменной x1 остается старая ссылка на отрезок зоны задач, хранящий число; изменяется лишь содержимое этого отрезка. Оператор позволяет экономить на поиске места в зоне задач, сохраняя новое значение по старому адресу.

9. "Выч(четное x1)" - x1 есть число в формате "целое со знаком" либо "длинное целое". Оператор истинен, если это число четное;

10. "Выч(программа x1 x2 x3)" - x1 есть набор псевдокоманд, определяющий вычисления для набора x2 вход-выходных данных "с плавающей запятой" и набора x3 вход-выходных данных "целое со знаком". Реализуется цикл этих вычислений. Вне о.д.з. оператор ложен. Каждая псевдокоманда представляет собой набор $(N\ n_1\ \ldots n_k)$, где N - номер операции; $n_1, \ldots, n_k$ - адреса операндов (входных и выходных). Каждая операция однозначно определяет типы своих операндов (т.е. "с плавающей запятой" либо "целое"). Поэтому адреса операндов суть

номера соответствующих разрядов набора x2 либо набора x3. Нумерация начинается с 1. Помимо вход-выходных ячеек, представленных в наборах x2 и x3, псевдокоманды могут использовать дополнительные ячейки. Номер дополнительной ячейки берется как произвольный неиспользуемый, однако он должен быть не более 200. Использование пакета псевдокоманд позволяет экономить на пересылках, возникающих при выполнении интерпретатором операторов ЛОСа. Рассматриваются следующие типы псевдокоманд (номер команды приведен в скобках после ее названия):

(a) Сложение в формате "с плавающей запятой" (1). Первые два операнда - слагаемые, третий - сумма. В остальных операциях порядок данных тот же: сначала идут входные ячейки, затем - выходные.
(b) Сложение в формате "целое со знаком" (2);
(c) Изменение знака числа в формате "с плавающей запятой" (3);
(d) Изменение знака числа в формате "целое со знаком" (4);
(e) Умножение в формате "с плавающей запятой" (5);
(f) Умножение в формате "целое со знаком" (6);
(g) Деление в формате "с плавающей запятой" (7);
(h) Деление с остатком для формата "целое со знаком" (8). Первые два операнда - делимое и делитель, затем идут неполное частное и остаток.
(i) Возведение в степень в формате "с плавающей запятой" (9);
(j) Экспонента в формате "с плавающей запятой" (10);
(k) Возведение в натуральную степень для формата "целое со знаком" (11);
(l) Модуль числа в формате "с плавающей запятой" (12);
(m) Модуль числа в формате "целое со знаком" (13);
(n) Натуральный логарифм (14). Здесь и далее все операции - формате "с плавающей запятой";
(o) Логарифм - общий случай (15);
(p) Квадратный корень (16);
(q) Синус (17), Косинус (18), Тангенс (19), Котангенс (20), Секанс (21), Косеканс (22);
(r) Арксинус (23), Арккосинус (24), Арктангенс (25), Арккотангенс (26);
(s) Гиперболический синус (27), Гиперболический косинус (28), Гиперболический тангенс (29), Гиперболический котангенс (30);
(t) Сигнум (31);
(u) Тождественная операция (32). Используется для присвоения значения выходной переменной;
(v) Нахождение целой части (33).

9.3.8 Вспомогательные процедуры для вычислений в форматах "с плавающей запятой" и "целое со знаком"

Чтобы вычислить значение константного терма x1 в формате "с плавающей запятой", используется операторное выражение "вычконст". Его программа имеет рекурсивный характер и использует обращения к операторному выражению "выч(*ldots*)". Предполагается, что терм x1 имеет только элементарные операции, реализуемые этим выражением.

Роль компилятора для получения наборов "псевдокоманд", реализуемых оператором "Выч(программа ...)", играет процедура "вычпрог(x1 x2 x3)". Здесь x1 - компилируемое выражение, определяющее либо одно числовое значение, либо числовой вектор, либо числовую матрицу. Выходной переменной x2 присваивается набор псевдокоманд, позволяющий вычислять значение x1 в формате "с плавающей запятой". В случае вектора или матрицы выходные данные компилируемой программы образуют несколько расположенных подряд числовых значений (элементы матрицы перечисляются по строкам). Выходной переменной x3 присваивается набор вспомогательных констант, имеющих формат "с плавающей запятой". Вход-выходной набор программы x2 состоит из набора значений параметров выражения x1, упорядоченных по возрастанию номеров; далее идет набор констант x3, и в конце размещается результирующее значение либо набор результирующих значений. Эти значения перед обращением выбираются произвольно (например, полагаются равными нулю) и изменяются программой в процессе вычислений.

В заключение приведем еще две простых, но часто используемых процедуры. Первая из них - операторное выражение "числзнач(x1)" - переводит число x1 из формата "с плавающей запятой" к виду десятичного числа ЛОСа. Она обеспечивает преобразование пары (десятичное число - десятичный порядок), выдаваемой оператором "выч(выход x1)", в одно десятичное число. Вторая процедура - оператор "смцелые(x1 x2 x3)", выполняющий перечисление целых чисел x3 начиная с целого x1 и кончая целым x2. Форматы чисел - "целое со знаком". Компилятор ГЕНОЛОГа использует данный оператор для организации циклов в вычислениях "с плавающей запятой".

9.4 Приемы сканирования задач, реализованные на ЛОСе

Перейдем к рассмотрению немногих приемов элементарной алгебры, реализованных на ЛОСе. Как правило, эти приемы относятся к константным выражениям и несложны. Исключение составляет лишь аппарат, обеспечивающий переход к новым неизвестным при решении уравнений и неравенств. Рассматриваемые ниже программы можно найти в разделе ("Приемы решателя", "Элементарная алгебра") оглавления программ.

Усмотрение истинности либо ложности строгого неравенства для констант

Если при сканировании задачи встречается строгое неравенство $a < b$, не содержащее переменных, то предпринимается попытка усмотреть его истинность либо ложность с помощью оператора "числоценка". Для усмотрения истинности определяются численные оценки c, d выражений a, b. Первая из них берется с избытком, вторая - с недостатком. Если $c < d$, то неравенство заменяется на константу "истина". Аналогичным образом реализуется попытка усмотреть ложность неравенства. Сначала вычисления проводятся с точностью двух знаков после запятой. Если этого не хватает, предпринимается повторная попытка с 4 знаками. Если и она не дает результата, вводится комментарий (числоценка a b), блокирующий дальнейшие попытки применения данного приема. Уровень срабатывания приема равен 0.

Усмотрение истинности либо ложности нестрогого неравенства для констант

Аналогично случаю строгих неравенств.

Определение знаменателя числовой дроби

Если в задаче встречается выражение "знаменатель(A)", где A - десятичная запись либо числовая дробь, то с помощью оператора "простаядробь(...)" находится знаменатель Q несократимой простой дроби, равной A, и рассматриваемое выражение заменяется на десятичную запись Q. Уровень срабатывания равен 0.

Определение числителя числовой дроби

Аналогично предыдущему.

Преобразование к стандартному виду дробной степени натурального числа

Если в задаче встречается степенное выражение A, основанием m которого служит натуральное число, а показатель степени, имеющий заголовок "дробь", равен простой несократимой дроби k/n, то применяется процедура "стандстепень(x1 x2 x3 x4)", обеспечивающая стандартизацию A. Здесь x1 - терм A; x2, x3 - числа k, n. Выходной переменной x4 присваивается набор сомножителей результата преобразований. Программа оператора выполняет следующие действия. Переменной x5 присваивается число m. Оно разлагается в произведение степеней простых чисел; x6 становится равно набору этих чисел, а x7 - набору их показателей степени. Выполняется просмотр пар (простое число p - его показатель степени r). С учетом того, что число m возводится в степень k/n, выделяется целая часть u показателя при p: $rk/n = u + v/n$. Элементы r заменяются в наборе x7 на величины v, а целочисленные степени p^u перемножаются. В тех случаях, когда степень слишком велика для непосредственного вычисления, она заносится в виде терма в накопитель x5. Иначе - находится результат возведения в степень, и переменной x8 присваивается произведение всех этих результатов. По окончании цикла выделения целочисленных множителей - переход через "иначе 2". Здесь выполняется группировка сомножителей $p^{v/n}$ под общие знаменатели показателей степени. Находится результат v'/n' сокращения дроби v/n. Знаменатели n' заносятся без повторений в накопитель x9, а пары (p, v') - в соответствующие списки накопителя x10. В каждой группе таких пар, соответствующей заданному значению n', определяется наибольший общий делитель w всех числителей v'. Далее целочисленные степени $p^{v'/w}$ вычисляются, находится произведение h всех этих степеней, и в накопитель результата x11 заносится сомножитель $h^{w/n'}$. Ранее в x11 были помещены термы списка x5 и десятичная запись числа x8. Прием срабатывает на уровне 0.

Представление в виде степени натурального числа числового основания степенного выражения

Если в задаче встречается выражение вида n^p, где n - десятичная запись натурального числа, то предпринимается попытка усмотреть в n натуральную степень другого натурального числа: $n = m^k$. Для этого применяется разложение n в произведение степеней простых чисел и определение наибольшего общего делителя показателей степеней. В случае успеха исходное выражение преобразуется к виду m^{kp}. Так как в решателе показатели степени обычно преобразуются к виду суммы выражений, перед заменой терм kp обрабатывается нормализатором раскрывания скобок "стандплюс". Если выражение входит в условие задачи на описание, причем показатель p содержит логарифм либо не известен, то уровень срабатывания приема равен 0, иначе - равен 2. После применения приема могут оказаться возможными усмотрение другого вхождения неизвестной степени с тем же основанием m и выбор степенного выражения вида m^A в качестве новой неизвестной. Поэтому сбрасывается комментарий (новыенеизвестные ...), блокирующий повторение попытки перехода к новым неизвестным.

Нормализация числового коэффициента при степени десяти

При завершающей обработке ответа геометрических задач "на вычисление" и других аналогичных задач (физика, химия и т.п.) обычно используется вспомогательная задача на преобразование, имеющая цель "учетрезультата". Появление в условии этой задачи выражения $A \cdot 10^n$, где A, n - десятичные записи, вызывает срабатывание приема, обеспечивающего такое изменение десятичного порядка n, чтобы величина A оказалась заключенной между 1 и 10. Уровень срабатывания равен 3.

Округление десятичной константы при редактировании ответа

В особых случаях решение задачи на преобразование, имеющей цель "учетрезультата", сопровождается округлением числовых констант, представляющих собой коэффициент ответа либо сам ответ. Пока такое округление производится только для задач, имеющих понятия из раздела "химия". Находится первая ненулевая цифра после запятой и выполняется округление до следующей после нее цифры.

Стандартизация логарифма рационального числа по рациональному основанию

Если в задаче встречается выражение $\log_a b$, где a, b - рациональные константы p/q, r/s, то предпринимается попытка его стандартизации с помощью процедуры "стандлогарифм(x1 x2 x3 x4 x5)". Здесь x1, x2 - числа p, q; x3, x4 - числа r, s. Выходной переменной x5 присваивается результирующий терм.

Программа процедуры выполняет следующие действия. Предпринимаются попытки найти такие максимальные натуральные k_1, k_2, что p/q есть k_1 - я степень какой-либо рациональной константы, а r/s - k_2 - я степень. Для этого используются разложения в произведения степеней простых чисел. Переменной x6 присваивается пара (k_1, k_2); из констант p, q извлекается корень k_1 - й степени, а из констант r, s - корень k_2 - й степени. Затем - переход через "иначе 2". Здесь переменной x7 присваивается такое целое число t, что r/s представляется в виде $r'/s' \cdot (p/q)^t$ для наименьших натуральных r', s'. Происходит переприсвоение $s := s', r := r'$. Значением переменной x10 становится терм R вида $\log_{p/q} r/s$. Если число под логарифмом равно 1, то этот терм заменяется на 0. Кроме того, предпринимается рекурсивное обращение к процедуре "стандлогарифм"для дальнейшего упрощения R. Наконец, формируется окончательный ответ $k_2(t+R)/k_1$. Уровень срабатывания приема равен 0.

Стандартизация линейной комбинации логарифмов натуральных чисел по общему основанию

Пусть в задаче встречается сумма, сомножителем слагаемого которой является логарифм натурального числа по некоторому, не обязательно константному, основанию c. Тогда в этой сумме выделяются все слагаемые вида $a_i \log_c n_i$; $i = 1, \ldots, k$, где a_i, n_i - целые. Переменная x13 играет роль накопителя вхождений выделенных слагаемых; переменная x14 - накопителя простых делителей чисел n_i; переменная x15 - накопителя показателей степеней, с которыми эти простые делители войдут в произведение чисел $n_i^{a_i}$. Для удобства программирования элементы набора x15 представляют собой одноэлементные наборы, составленные из показателей степени. По завершении заполнения накопителей x13-x15 выполняется переход через "иначе 2". Здесь проверяется неравенство $k > 1$. Затем переменной x16 присваивается пара наборов A_1, A_2. Первый из них образован всевозможными парами (простое число - показатель степени) для всех положительных показателей степени, имеющихся в x15; второй - всевозможными парами (простое число - показатель степени с измененным знаком) для отрицательных показателей. По каждой паре создается свое логарифмическое выражение $d \log_c r$. Здесь d - наибольший общий делитель показателей степени; r - результат перемножения первых элементво пар, возведенных в соответствующие степени, сокращенные на d. Наконец, выделенные слагаемые заменяются на разность указанных логарифмических выражений. Уровень срабатывания приема равен 0.

Стандартизация логарифма с основанием, представимым в виде степени натурального числа

Если в задаче встречается выражение $\log_n a$, где n - натуральная константа, то эта константа разлагается в произведение степеней простых множителей. Определяется наибольший общий делитель d их показателей степени (переменная x9). Если $d \neq 1$, находится результат m перемножения простых сомножителей, возведенных в степени, сокращенные на d. Затем логарифм заменяется на $\log_m a/d$. Уровень срабатывания равен 1.

Логарифм натуральной степени натурального числа

Аналогично предыдущему, но анализируется натуральная константа под логарифмом.

Переход от отношения "больше" к отношению "меньше"

Решатель не использует отношения "больше" - оно сразу же, на уровне 0, переформулируется через "меньше". Исключение составляют задачи на анализ текста, где выражение "больше" может сохраняться до завершения семантического анализа.

Подстановка в условие задачи на доказательство выражения, извлекаемого из линейного соотношения в посылках

Если в посылках задачи на доказательство усматривается равенство вида $a_1 + \ldots + a_n = 0$ (вещественное либо комплекснозначное), причем некоторое a_i содержит все параметры, входящие в прочие слагаемые $a_1, \ldots, a_{i-1}, a_{i+1}, \ldots, a_n$, а также хотя бы один параметр, не входящий в эти слагаемые, то предпринимается замена вхождений a_i в условия той же задачи на сумму остальных слагаемых с измененными знаками. Прием срабатывает на уровне 2; за одно его срабатывание заменяется лишь одно вхождение a_i в условие.

Попытка перейти к новым неизвестным при рассмотрении числовых уравнений в задаче на описание

Если решается система численных уравнений $A_1, \ldots, A_n$ относительно неизвестных $x_1, \ldots, x_n$, то может оказаться полезным переход к новым неизвестным $y_1, \ldots y_n$. Для выполнения этого перехода предпринимается попытка усмотреть такие выражения $t_1, \ldots, t_n$, что каждое вхождение неизвестной в $A_1, \ldots, A_n$ встречается только внутри одного из них. Иногда уравнения приходится предварительно преобразовать, чтобы добиться точных повторений одного и того же выражения в разных местах. После того, как выражения $t_1, \ldots, t_n$ найдены, выбираются новые переменные $y_1, \ldots, y_n$, уравнения A_i относительно неизвестных x преобразуются в уравнения A'_i относительно переменных y и решаются. В найденном ответе R неизвестные y_i заменяются на выражения t_i, и полученное утверждение R' замещает в текущей задаче условия $A_1, \ldots, A_n$.

Вся эта достаточно сложная работа выполняется приемом, реализованным на ЛОСе. Прием использует две вспомогательные процедуры. Первая из них - пакетный нормализатор "повторчисло". Он реализован на ГЕНОЛОГе и будет описан в последующих разделах. Нормализатор выполняет преобразования уравнений, направленные на явное получение повторяющихся подвыражений с неизвестными. Вторая процедура - "новыенеизвестные". Она реализована на ЛОСе и предназначена для усмотрения новой подсистемы, число уравнений которой равно числу новых неизвестных.

Перейдем к рассмотрению программы приема, содержащейся в разделе ("Приемы решателя", "Элементарная алгебра", "Попытка перейти к новым неизвестным при рассмотрении числовых уравнений в задаче на описание") оглавления программ.

Работа приема инициируется при обнаружении корневого равенства в условии задачи на описание. Уровни срабатывания приема равны 0, 2 и 6. Фактически уровни 0 и 2 альтернативные. Первоначально попытка применить прием происходит на уровне 0; при этом сразу вводится комментарий, блокирующий его применение на уровне 2. Последующие приемы могут выполнять разблокировку срабатывания данного приема на уровне 0 либо на уровне 2 - в зависимости от ситуации. Если нужна пауза перед попыткой перехода к новым неизвестным, в течение которой успели бы сработать другие приемы, разблокируется уровень 2, иначе - уровень 0. Попытка срабатывания на уровне 6 - контрольная, выполняемая по исчерпании прочих средств. К этому моменту уравнения могут оказаться преобразованы таким образом, что возможность ввода новых неизвестных будет легко усматриваться.

Повторные попытки срабатывания блокируются комментариями (новыенеизвестные u), где u - один из указанных трех уровней срабатывания. Для каждого уровня вводится свой комментарий. Некоторые другие приемы, после срабатывания которых становится возможен переход к новым неизвестным, удаляют ранее введенные комментарии (новыенеизвестные ...), активируя таким образом повторные попытки перехода.

Прежде всего, прием анализирует целевую установку задачи, отсекая случаи, когда нет надобности в его применении. Так как один из уровней срабатывания равен 0, проверяется отсутствие конъюнкции в условиях: под конъюнкцией может оказаться "спрятана" часть уравнений, и нужно подождать, пока не сработает прием, устраняющий ее. Проверяется отсутствие комментария (новыенеизвестные стоп), блокирующего применение данного приема согласно каким-либо внешним соображениям. Проверяется, что хотя бы один из операндов равенства имеет своим заголовком операцию, принимающую числовые значения и что равенство содержит неизвестные, т.е. является уравнением. Проверяется отсутствие в уравнении описателей. Наличие цели (связка ...) указывает на работу с функциональными (например, дифференциальными) уравнениями и тоже отсекается. Проверяется отсутствие дизъюнктивного условия с неизвестными: по этому условию сначала будет выполняться разбор случаев. Наконец, проверяется, что рассматриваемое уравнение не является равенством, выражающим значение неизвестной через известные параметры.

После перечисленных проверок вводится комментарий (новыенеизвестные x5). Если x5 равно 0 и не было комментария (новыенеизвестные 2), то он вводится. Кроме того, на уровне 0 вводится комментарий (новыенеизвестные исследовать). Он блокирует срабатывание аналогичного приема для блока анализа задачи на описание. Эта блокировка будет снята, если в блоке анализа окажутся выведены какие-либо новые уравнения. Заметим, что попытка перехода к новым неизвестным инициируется рассмотрением какого-либо одного уравнения. Прочие уравнения будут рассматриваться при уже введенной блокировке, отсекающей прием.

Начинается цикл анализа уравнений задачи. Переменной x6 присваивается список всех имеющих неизвестные числовых равенств, находящихся в списке условий. Вводится накопитель x7 уравнений, получающихся из уравнений x6 предварительными преобразованиями. Эти преобразования позволяют выделить в явном виде возможно большее количество повторяющихся вхождений одинаковых подвыражений с неизвестными. Изначально x7 есть копия списка x6. Вводится накопитель x8 комментариев (выводимо ...), (коррекцияпосылок ...), которые будут сопровождать уравнения списка x7. Они указывают вспомогательные утверждения, использованные при преобразованиях уравнения, а также необходимые для сопровождения по о.д.з. утверждения, относящиеся к новым выражениям с неизвестными, появившимся после преобразований. Переменной x9 присваивается контекст преобразований - объединение списка посылок задачи со списком всех ее условий, отличных от уравнений.

Далее реализуется цикл преобразований. Он состоит из многократных просмотров набора x7. Переменная x10 представляет собой подсписок фактически просматриваемых на текущей итерации элементов этого набора. Первоначально x10 полагается равным x7. В процессе просмотра уравнений и их преобразований заполняется накопитель x11 тех элементов списка x7, к которым нужно будет вернуться на следующей итерации. Он и будет определять новое значение списка x10. В процессе просмотра переменная x12 указывает на текущее вхождение уравнения в список x7; переменная x13 - на соответствующее вхождение в список комментариев x8. Значением переменной x14 служит текущее уравнение. Проверяется, что оно входит в "активный" подсписок x10. Затем вместо этого уравнения в список x7 временно помещается 0. В результате x7 становится указателем "внешнего контекста" уравнения x14 среди прочих уравнений. Он будет передаваться процедуре "повторчисло" , учитывающей (наряду с x14) данный контекст для получения повторных вхождений тех выражений, которые в нем уже имеются. Эта же процедура будет принимать решения о повторном рассмотрении других уравнений.

Переменным x15 и x16 присваиваются комментарии (выводимо ...), (коррекцияпосылок ...), необходимые для работы с x14. При первом просмотре списка x7 создаются исходные версии комментариев, имеющие пустые накопители. На последующих просмотрах извлекаются уже имеющиеся в списке x8 экземпляры данных комментариев. Переменной x17 присваивается список посылок, используемых при преобразовании уравнения x14. Он получается из списка x9 добавлением всех утверждений, необходимых для сопровождения по о.д.з. текущей версии уравнения. Эти утверждения берутся из комментария (коррекцияпосылок ...). Переменной x18 присваивается комментарий (новыенеизвестные пустоеслово), используемый как накопитель уравнений, подлежащих повторному рассмотрению. Наконец, реализуется обращение к пакетному нормализатору "повторчисло". Формат обращения - стандартный для нормализаторов: первый аргумент представляет собой преобразуемый терм x14; второй - список посылок x17; третий - список комментариев. Кроме уже упомянутых выше комментариев x15, x16 и x18, в него заносятся цель (неизвестные ...) текущей задачи и элемент (внешвхождение x7), определяющий список остальных уравнений. Если уравнение было единственным, элемент не создается.

Приемы оператора "повторчисло" реализованы на ГЕНОЛОГе. Мы рассмотрим их позднее. Заметим, что преобразование исходных уравнений для выявления повторяющихся подвыражений может выполняться неоднозначно. Однако, вместо сравнительно трудоемкого перебора всех таких преобразований, в решателе реализована процедура последовательных приближений к некоторому единственному "эвристически наилучшему" варианту. Она работает быстрее, и на практике оказалась вполне достаточна.

По завершении цикла преобразований уравнений - переход к фрагменту, начинающемуся с контрольной точки "прием(20)". Здесь происходит просмотр преобразованных уравнений и составление списка x12 указателей повторяющихся вхождений подвыражений с неизвестными. В конечном счете, x12 оказывается состоящим из пар (a - M), где a - заголовок отличных от переменных неизвестных подвыражений, встречающихся неоднократно; M - список пар (неизвестное подвыражение t с заголовком a - список пар (уравнение, содержащее t - вхождение t в это уравнение)). Сначала в список x12 заносятся все данные о неизвестных подвыражениях, затем - отбрасываются подвыражения, встречающиеся однократно.

Переменной x13 присваивается список всех переменных, встречающихся в текущей задаче на описание; переменной x14 - список ее неизвестных; переменной x15 - число неизвестных. После контрольной точки "прием(21)" начинается отбор подсистемы уравнений, преобразуемых к новым неизвестным. Последовательно фиксируются значения указателя x16 числа уравнений подсистемы. Сначала предпринимается попытка усмотреть одно уравнение с одной новой неизвестной, затем - два уравнения с двумя неизвестными, затем - три уравнения с тремя. Для текущего значения x16 проверяется, что либо оно не больше x15, либо задача имеет хотя бы x16 уравнений. Переменной x17 присваивается пара (стоп пустоеслово), второй элемент которой будет играть роль накопителя уже просмотренных уравнений списка x7.

Далее просматриваются модифицированные уравнения x19 списка x7, и для каждого такого уравнения предпринимается обращение к процедуре "новыенеизвестные(x19 x12 x18 x20)". Здесь x18 - список информационных элементов, содержащий наборы x14, x17, а также пары (переменные x13) и (число x16). Эта процедура предпринимает попытку найти группу из x16 уравнений $A_1, \dots, A_n$, упоминаемых в списке x12 и включающую уравнение x19, для которой возможен переход к x16 новым неизвестным. Выходная переменная x20 перечисляет четверки (набор уравнений $(A_1, \dots, A_n)$ - набор выделенных в этих уравнениях неизвестных подвыражений $t_1, \dots, t_n$ - набор $(B_1, \dots, B_n)$ уравнений, полученных из $A_1, \dots, A_n$ заменой подвыражений $t_1, \dots, t_n$ на новые неизвестные $y_1, \dots y_n$ - набор переменных $(y_1, \dots, y_n)$). Предполагается, что каждая неизвестная встречается в уравнениях $A_1, \dots, A_n$ только внутри выражений $t_1, \dots, t_n$. Вообще говоря, часть выражений t_i могут быть неизвестными текущей задачи. Программа процедуры "новыенеизвестные" не содержит каких-либо принципиально интересных моментов - она просто реализует несложный перебор.

Проверяется, что все выражения t_i для новых неизвестных принимают числовые значения. Иногда имеет смысл составить систему, имеющую больше уравнений, чем исходная задача имела неизвестных. Однако, здесь предусмотрен контроль вырожденных случаев - проверяется отличие каждого t_i от переменной и существование хотя бы двух вхождений в новые уравнения каждой новой неизвестной.

Выполняется просмотр списка выражений t_i и упрощение их при помощи пакетного нормализатора "упрощповтор", учитывающего типичные ситуации, возникающие при формировании термов t_i оператором "повторчисло".

Отбрасывается ряд неинтересных случаев. Во-первых, проверяется существование новой неизвестной, не расположенной в новых уравнениях под тригонометрической функцией. Этим отсекаются попытки тривиальных переобозначений неизвестных аргументов тригонометрических функций. Если каждое выражение t_i есть либо неизвестная, либо степенное выражение с неизвестной в основании и известным показателем степени, то проверяется, что число степенных выражений более одного. Если некоторое t_i содержит неизвестное тригонометрическое выражение, то проверяется отсутствие неизвестных тригонометрических выражений в новых уравнениях.

Если новая система уравнений отброшена, то уравнение x19 регистрируется в накопителе x17, и продолжение просмотра списка x7. Те уравнения, которые помещены в x17, процедурой "новыенеизвестные" не рассматриваются. Если же система уравнений не отброшена, то создается вспомогательная задача на описание x24. Ее посылками становятся все посылки текущей задачи, а также все не содержащие неизвестных условия текущей задачи. Условиями служат новые уравнения $B_1, \dots, B_n$, к которым добавлены утверждения "число(y_1)", ..., "число(y_n)". Неизвестные суть переменные $y_1, \dots, y_n$. Задача имеет цели "одз", "полный", "явное", "прямойответ", "новыенеизвестные".

Реализуется цикл учета информационных элементов (выводимо ...), (коррекцияпосылок ...), извлекаемых из списка x8 для отобранных уравнений A_i списка x7. Те утверждения, которые нужны для сопровождения по о.д.з. подвыражений уравнений A_i, преобразуются к новым неизвестным y_j, чтобы обеспечивать сопровождение по о.д.з. для уравнений B_i. Они добавляются к условиям задачи x24. Вводится список x23, состоящий из исходных версий уравнений A_i. В нем регистрируются некоторые другие условия, которые станут избыточными после регистрации в текущей задаче ответа вспомогательной задачи x24. Далее рассматриваются комментарии (сопровождение ...) к текущей задаче. Утверждения, сопровождающие по о.д.з. подвыражения уравнений A_i, тоже выражаются через новые неизвестные и добавляются к условиям задачи x24.

Наконец, формирование задачи x24 завершено, и после контрольной точки "прием(2 4)" выполняется обращение к ее решению. Если получен "отказ", то вводится комментарий (новыенеизвестные стоп) к текущей задаче, блокирующий повторные попытки перехода к новым неизвестным, и продолжение сканирования. Иначе - находится результат x27 подстановки в найденный ответ x25 выражений $t_1, \ldots, t_n$ вместо новых неизвестных $y_1, \ldots, y_n$. Просматриваются условия текущей задачи, вошедшие в список x23. Те из них, которые нужны для сопровождения по о.д.з., сохраняются, прочие - удаляются. Конъюнктивные члены утверждения x27 заносятся в список условий текущей задачи. Те из них, которые содержат квантор существования, помечаются комментарием "серия". Комментарий (выводимо ...) к задаче x24 используется для регистрации в текущей задачи информации об использованных посылках. Анализируется комментарий (сопровождение ...) к задаче x24. В тех его элементах, которые относятся к сопровождению по о.д.з. подвыражений ответа, происходит исключение новых неизвестных, и эти элементы регистрируются в комментарии (сопровождение ...) к текущей задаче. В заключение прием удаляет комментарий (новыенеизвестные 0), чтобы разблокировать попытку перехода к новым неизвестным в изменившейся ситуации. В нескольких задачах такая попытка была необходима.

Попытка перейти к новым неизвестным при рассмотрении числовых уравнений в задаче на исследование

Даже если поначалу переход к новым неизвестным в задаче на описание не усматривается, он может оказаться возможным в процессе вывода следствий из исходных уравнений. Такой вывод следствий (линейные комбинации уравнений, деление и умножение их, возведение в степень, и т.п.) предпринимается в посылках блока анализа задачи на описание. Чтобы здесь усматривать возможность перехода к новым неизвестным, создан прием, аналогичный предыдущему. Так как число уравнений в блоке анализа обычно бывает достаточно большим, трудоемкость процедуры заметно повышается.

Инициализация приема происходит на уровне 1 при рассмотрении корневого вхождения равенства. Прежде всего, анализируется целевая установка задачи. В задачах, имеющих цель "известно", прием не применяется. Здесь применяются другие способы работы с уравнениями, о которых будет рассказано в разделе, посвященном элементарной геометрии. При отсутствии цели "известно" прием блокируется, если имеется указывающая на геометрическюу задачу посылка вида "точка(A)".

Задача на исследование не всегда является блоком анализа задачи на описание (например, может быть нужна для усмотрения противоречивости списка посылок), поэтому проводится проверка существования внешней задачи на описание. В рамках последней уже имелась попытка перехода к новым неизвестным. Поэтому новые попытки будут предприниматься только для новых уравнений, по мере их появления. Для указания на то, что рассмотрение посылки уже выполнялось, используется ее комментарий "новыенеизвестные".

Если работа с блоком анализа только началась, что обнаруживается по отсутствию общего комментария "новыенеизвестные" к посылкам, то блокируется рассмотрение исходных уравнений. Для этого прием сопровождает каждое (в том числе текущее) уравнение задачи комментарием "новыенеизвестные", вводит комментарий "новыенеизвестные" к списку посылок, и его работа завершается.

Если рассматриваемое равенство - действительно новое, то проверяется, что его части принимают числовые значения и что оно содержит неизвестные. Блокируется рассмотрение равенств, имеющих функциональные подвыражения "значение(f t)".

Затем переменной x6 присваивается список всех новых (не имеющих комментария "новыенеизвестные") уравнений задачи, а переменной x7 - список всех старых уравнений.

Если среди уравнений задачи имеется равенство тригонометрической функции неизвестного выражения известному выражению, то прием блокируется, так как его срабатывание представляется преждевременным (текущий уровень равен 1, а приемы, решающие простейшие тригонометрические уравнения, срабатывают на более высоких уровнях).

Далее все уравнения списка x6 помечаются комментарием "новыенеизвестные". С этого момента действия аналогичны случаю уравнений задачи на описание. Вводится список x8 - накопитель результатов преобразований уравнений к виду с повторяющимися вхождениями подвыражений. Изначально он равен конкатенации списков x6 и x7. Вводится накопитель x9 информационных элементов (коррекцияпосылок ...), (выводимо ...), сопровождающих преобразованные уравнения.

Реализуется цикл обращений к оператору "повторчисло". Роль накопителя уравнений, к рассмотрению которых нужно вернуться на следующей итерации, играет переменная x12. Кроме комментариев, упоминавшихся выше для случая задачи на описание, оператору "повторчисло" передается также комментарий "исследовать". По завершении цикла формируется указатель x13 повторных вхождений неизвестных подвыражений. Далее применяется процедура "новыенеизвестные", формируется вспомогательная задача, посылками которой становятся все не содержащие неизвестных посылки задачи на исследование, реализуется обращение к ее решению, и найденный ответ замещает исходные уравнения. Все эти действия аналогичны случаю задачи на описание.

Попытка перейти к новым неизвестным при решении неравенств

В случае неравенств созданы два отдельных приема - для строгих и нестрогих неравенств. Они практически ничем друг от друга не отличаются, однако инициируются различными логическими символами - "меньше" и "меньшеилиравно". По сравнению с уравнениями, ситуация здесь более простая - преобразуется лишь одно неравенство, которое разрешается относительно единственной новой неизвестной. Приемы применяются к условию задачи на описание; уровень срабатывания их равен 0. Прежде всего, анализируется целевая установка задачи и отсекаются случаи, когда попытка перехода к новым неизвестным нецелесообразна. В частности, проверяется отсутствие уравнений с неизвестными, входящими в неравенство. Решение уравнений является более приоритетным, так как может избавить от решения неравенств, сведя их рассмотрение лишь к проверке. Проверяется также, что неравенство не используется для сопровождения по о.д.з. Дальнейшие действия представляют собой упрощенный вариант действий, рассмотренных выше для уравнений.

Попытка решения уравнения путем получения качественного описания графика разности его частей и подбора корней

Иногда можно найти корни численного уравнения методом угадывания нескольких значений, удовлетворяющих этому уравнению, и доказательства отсутствия прочих корней. Последнее обычно основано на анализе промежутков монотонности разности частей уравнения. При угадывании корней берутся либо малые целочисленные значения, либо такие константы, при которых пропадают какие-либо сложные фрагменты уравнения (например, логарифмические выражения, тригонометрические функции и т.п.). Попытка применить прием, выполняющий эти действия, инициируется на уровне 10 при рассмотрении текущего символа "равно" - корня условия задачи на описание. Проверяется, что задача имеет единственную неизвестную, входящую в данное условие и что условие не имеет других переменных. Проверяется, что части уравнения принимают численные значения и что оно имеет хотя бы одну "сложную" операцию - логарифм, тригонометрическую функцию либо экспоненту. Проверяется, что неизвестная принимает численные значения. Для блокировки повторных применений приема используется комментарий условия "возрастает".

Прием использует обращение к вспомогательной задаче на качественное исследование графика функции $\lambda_x(f(x), x - \text{число})$, где $f(x)$ - разность частей уравнения. Такие задачи решаются с помощью приемов математического анализа и будут рассмотрены в последующих разделах. Тип вспомогательной задачи - "описать"; ее условием служит равенство $y = \lambda_x(f(x), x - \text{число})$, где y - новая переменная. Задача имеет цели "полный", "явное", "прямойответ", "(неизвестные y)", "исследовать", "функция". Ответом задачи служит конъюнкция утверждений, характеризующих рассматриваемую функцию: указывающих ее область определения, промежутки монотонности, точки экстремумов и число корней на промежутках монотонности. Прием выбирает из этих утверждений равенство, определяющее область определения M функции y, а также группу утверждений, определяющих количества корней на промежутках монотонности. Находится множество x23 точек области M, не покрытых такими промежутками, и проверяется, что оно либо пусто, либо состоит из конечного числа явно определенных точек.

Затем просматриваются промежутки с ненулевым числом корней, и с помощью оператора "подборкорня" осуществляется генерация проверяемых на роль корней констант. Пока реализована простейшая версия данного оператора. Она пытается исключить логарифмы $\log_a x$ по константному основанию a выбором для x значения a^n, где n - целое, не превосходящее по модулю 6. Затем перебираются целочисленные значения, не превосходящие по модулю 4. После проверки того, что текущее значение расположено на рассматриваемом промежутке и удовлетворяет уравнению, оно регистрируется в накопителе x16. По завершении цикла просмотра промежутков формируется заменяющее утверждение $x = a_1 \ \vee \ \dots \ \vee x = a_n \vee (A = B \ \& \ (x = b_1 \ \vee \dots \ \vee \ x = b_m))$. Здесь $A = B$ - исходное уравнение; $a_1, \dots, a_n$ - найденные корни; $b_1, \dots, b_m$ - особые точки списка x23.

Усмотрение нецелочисленности константы

Для усмотрения ложности утверждений вида "целое(a)", где a - константный терм, используется обращение к процедуре "числоценка". Находятся оценки b_1, b_2 величины a с недостатком и с избытком; уровень точности - три знака после запятой. Если b_1, b_2 не целые, но имеют одну и ту же целую часть, то указанное утверждение заменяется на константу "ложь". Прием срабатывает на уровне 0.

Вычисление значения выражения с заданным числом знаков после запятой

В задаче на преобразование, имеющей цель (числоценка n), требуется найти приближенное значение выражения с точностью до n знаков после запятой. Условием такой задачи служит константное числовое выражение. Прием обращается к процедуре "числоценка" для получения оценок с недостатком и избытком до тех пор, пока результаты округления этих оценок до n знаков после запятой не совпадут. После очередной неудачи количество знаков, с которым проводятся вычисления, увеличивается на 1, но не более 6 раз. По превышении этого лимита (ситуация с безразличным округлением) берется результат, соответствующий меньшей оценке. Уровень срабатывания приема равен 1.

Переход от неизвестных к параметрам, если относительно них система уравнений линейна

Прием ориентирован на крайне редкие случаи, когда система уравнений нелинейна относительно неизвестных, но линейна относительно численных "известных" параметров. В этом случае предпринимается попытка разрешить сначала уравнения относительно данных параметров, а затем решать полученную систему относительно исходных неизвестных. Уровень срабатывания равен 8, т.е. существенно выше уровня стандартных приемов решения уравнений. Находится список x6 всех числовых уравнений в условиях задачи на описание и проверяется, что число неизвестных в этих уравнениях равно числу уравнений. Проверяется, что система нелинейна. Находится список x9 всех "известных" параметров, относительно которых уравнения системы линейны. Проверяется, что число уравнений равно числу таких параметров и что каждый из них имеет хотя бы два вхождения в систему. Затем создается вспомогательная задача на описание x13 с неизвестными x9. Ее условия получены из условий текущей задачи добавлением всех посылок, имеющих вхождения переменных списка x9. После решения задачи x13 ее условия замещают условия текущей задачи. Повторно вводится цель "одз", позволяющая скорректировать сопровождение по о.д.з.

Разделение неизвестных в системе неравенств с общим функциональным выражением

Рассматривается задача на описание, имеющая цель "пример", у которой каждая неизвестная является несущественной. Некоторое условие этой задачи содержит подвыражение "значение($f\ x$)", причем x - неизвестная. Проверяется, что каждое содержащее x условие задачи - либо простейшее утверждение $P(x)$, где P - символ одноместного отношения, либо неравенство, у которого каждое вхождение переменной x расположено внутри терма "значение($f\ x$)". Проверяется, что эти вхождения переменной x не являются связанными. Находится список x9 всех неравенств с переменной x и проверяется, что он имеет хотя бы два элемента. С помощью пакетных операторов "усмнеубывает", "усмневозрастает" проверяется, что истинность неравенств списка x9 монотонно зависит от изменения $f(x)$: если неравенство истинно при некотором конкретном значении $f(x)$ и это значение увеличивается (уменьшается), то истинность сохраняется. В этой ситуации нет необходимости обеспечивать истинность всей системы неравенств x9 для одного общего значения x. Достаточно для каждого из них определить свое значение x, а затем взять максимум (минимум) этих значений. Поэтому прием выбирает новые переменные $x_1, \dots x_n$, по числу неравенств x9, заменяет в этих неравенствах $f(x)$ на $f(x_i)$, заменяет каждое условие $P(x)$ на $P(x_1), \dots, P(x_n)$, после чего исключает несущественную неизвестную x и вводит вместо нее несущественные неизвестные $x_1, \dots x_n$. Уровень срабатывания приема равен 2.

Приемы для целочисленных констант

Если в задаче встречается выражение "нод(m n)" либо "нок(m n)", где m, n - десятичные записи целых чисел, то вычисляется значение этого выражения, и выполняется замена. Если встречается утверждение "простое(n)", где n - десятичная запись целого числа, то проверяется его простота, и утверждение заменяется на логическую константу.

9.5 Приемы пакетных операторов, реализованные на ЛОСе

Несколько пакетных операторов имеют несложные арифметические приемы, реализованные непосредственно на ЛОСе и аналогичные описанным выше приемам сканирования задачи. Дадим краткий перечень этих приемов.

1. Оператор "нормчислитель". Определяется числитель константной дроби.

2. Оператор "нормзнаменатель". Определяется знаменатель константной дроби.

3. Оператор "стандплюс". Этот оператор предназначен для раскрывания скобок и преобразования произведений тригонометрических функций в суммы. Для выполнения завершающей стандартизации результата он обращается к процедуре "общнорм", реализованной на ЛОСе. Последняя предпринимает попытку усмотреть в контексте применимое к преобразуемому выражению равенство, являющееся заведомо упрощающим, и использовать его. Заведомо упрощающими считаются равенства, заменяющая часть которых образует собственный фрагмент заменяемой, либо такие, которые неконстантное выражение преобразуют в константное, либо такие, у которых заменяющая часть образована из фргамента заменяемой отбрасыванием части операндов. Процедура "общнорм" может также выполнять некоторые простые преобразования выражений, задающих наборы.

4. Оператор "видумножение". Этот оператор предназначен для разложения на множители и сложения дробных выражений. На ЛОСе реализованы: прием для разложения целого числа в произведение степеней простых чисел и два приема, выполняющие стандартизацию логарифмов с константным основанием. Разложение на множители целого числа бывает необходимо лишь в специальных случаях, которые распознаются по наличию комментария "натуральное", вводимого при обращении к оператору "видумножение".

5. Оператор "уравнлогарифм". Оператор используется для стандартизации логарифмов с неизвестными. Кроме приемов ГЕНОЛОГа, в него введены два приема стандартизации логарифмов с константным основанием, реализованные на ЛОСе.

6. Оператор "нормлогарифм". Пакет общей стандартизации логарифмических выражений. На ЛОСе реализованы три приема: упрощение логарифмов, основание которых есть степень натурального числа, упрощение константных логарифмов, а также упрощение логарифмов чисел, представимых в виде натуральной степени натурального числа.

7. Оператор "нормстепень". Пакет общей стандартизации степенных выражений. На ЛОСе реализован прием, представляющий в виде степени натурального числа натуральное основание степени.

8. Оператор "нормцелаячасть". На ЛОСе реализованы два приема: определение целой части дробной константы и константного выражения общего вида. В последнем случае используется процедура "числоценка".

Глава 10

Приемы элементарной алгебры, реализованные на ГЕНОЛО́Ге

К разделу "элементарнаяалгебра" отнесены следующие логические символы: цифры от 0 до 9; "величина", "число", "Число", "минус", "плюс", "умножение", "дробь". Этот раздел имеет подразделы "неравенства", "модули", "целыечисла", "степени", "логарифмы", "тригонометрия", "гипфункции", "комбинаторныефункци", "многочлены". Понятия, содержащиеся в подразделах, перечислим ниже.

10.1 Приемы, связанные с цифрами

Переходим к разделу оглавления базы приемов ("Элементарная алгебра", "Цифры"). Здесь собраны несколько простых приемов, связанных с цифрой "0" и инициируемых при усмотрении нуля (кроме единственного приема, инициируемого при усмотрении символа "набор"). Здесь же находится один из центральных пакетных операторов элементарной алгебры - проверочный оператор "усмне0". Начнем с рассмотрения приемов сканирования задачи.

Попытка усмотреть ложность равенства без неизвестных в условиях задачи на описание

Если в условиях задачи на описание встречается равенство, не содержащее неизвестных, то относительно часто оно оказывается ложным. Например, это может получиться при разборе случаев, часть из которых оказываются нереализуемыми. Проверка ложности таких равенств выполняется специальным приемом. Заметим, что все ненулевые члены равенства без неизвестных обычно группируются в левой части. Поэтому проверке подлежат утверждения вида $a = 0$. Прием имеет заголовок "второйтерм" и основан на теореме вида:

$\forall_a(\neg(a = 0) \to \neg(a = 0))$

Его антецедент обращается к проверочному оператору "усмне0", подробно рассматриваемому ниже. Прием имеет фильтры "уровень(1)", "условие", "тип(описать)", "известно(x1)", "внешзнак(и или)". Последний фильтр означает, что если равенство не корневое, то внешними операциями должны служить лишь логические связки "и", "или". Антецедент выделен указателем "блокпроверок"; указатель "лимит(5000)" отводит для попытки проверки крайне малую трудоемкость.

Попытка усмотреть ложность константного равенства

Прием аналогичен предыдущему, однако проверка ложности равенства предпринимается в другой ситуации. Рассматривается равенство нулю константного выражения. Обычно такие равенства ложные. Прокомментируем фильтры приема. Уровень срабатывания равен 1. Отбрасываются случаи, когда равенство является условием задачи на доказательство либо преобразование, так как здесь оно чаще всего бывает истинным. Не анализируются посылки задачи на описание, имеющей цель "стандравно". Эта задача возникла для вспомогательных вычислений, нужных в некоторой задачи на исследование, и ее посылки взяты из последней, т.е. уже анализировались. Для задачи на исследование, имеющей цель "известно", проверка ложности корневого равенства предпринимается лишь при наличии цели "контроль". Такая цель означает, что имел место разбор случаев и разбирается некоторый подслучай. Она включает усиленный контроль противоречивости посылок.

Усмотрение ненулевого выражения

В разделе собраны несколько приемов, усматривающих истинность отрицания равенства нулю либо ложность такого равенства. Начнем с тех из них, которые основаны на уже приводившейся выше теореме:

$\forall_a(\neg(a = 0) \to \neg(a = 0))$

Таких приемов всего три. Они имеют указатель "прямойответ", означающий, что идентифицируется именно отрицание равенства. Антецедент обрабатывается проверочным оператором. Первый прием срабатывает на уровнях 2 и 4 в условии задачи. Если это условие относится к задаче на описание и содержит неизвестные, то вероятность его истинности меньше, так что уровень срабатывания равен 4, иначе он равен 2. Фильтр "проверка(не(сопровождение)1)" отбрасывает случаи, когда отрицание равенства используется для обеспечения сопровождения по о.д.з. Конструкция "проверка(... 1)" указывает на размещение программы фильтра перед обращением к проверочному оператору, отсекая таким образом ненужные вычислительные затраты. Второй прием срабатывает на уровнях 0 и 2. Он применяется к корневому отрицанию равенства - условию задачи на описание, имеющему неизвестное тригонометрическое подвыражение. Понижение уровня срабатывания форсирует отбрасывание ненужных отрицаний равенств нулю значений тригонометрических операций, использовавшихся до этого для сопровождения по о.д.з. Двойной уровень срабатывания позволяет перехватывать случаи, когда высвобождение отрицания равенства от сопровождения по о.д.з. произошло между первой и второй попытками применения приема. Наконец, третий прием применяется к конъюнктивным членам утверждения под описателем "отображение", выполняя таким образом стандартизацию задания функции. Указатель "сопровождение" разрешает применение этого приема независимо от того, используется ли преобразуемое утверждение для сопровождения по о.д.з.

Несколько приемов усматривают ложность равенства нулю из строгих неравенств, имеющихся в контексте. Теоремы их таковы: $\forall_a(0 < a \to \neg(a = 0))$, $\forall_a(a < 0 \to \neg(a = 0))$, $\forall_{ab}(0 < a-b \to \neg(b-a = 0))$. Все они имеют заголовок "второйтерм" и указатель "сопровождение", разрешающий замену даже в случае, когда утверждение используется для сопровождения по о.д.з. Это естественно, так как в контексте остается строгое неравенство, обеспечивающее для сопровождения ту же информацию.

Наконец, в разделе есть два приема, ориентированных на сравнительно редкие ситуации. Первый из них усматривает отличие от нуля суммы, имеющей степенное слагаемое:

$\forall_{abcd}(0 \leq d \ \& \ 0 < a \ \& \ 0 < b \to \neg(ab^c + d = 0))$

Здесь основание степени b представляет собой константу. Антецеденты обрабатываются проверочными операторами, которым отводится крайне малый лимит трудоемкости. Уровень срабатывания нулевой. Второй прием использует имеющееся в списке посылок равенство нулю суммы, основанием степени слагаемого которой служит сравниваемое с нулем выражение:

$\forall_{abcd}(a + b^d c = 0 \ \& \ \neg(a = 0) \to \neg(b = 0))$

Первый антецедент берется непосредственно из посылок, второй обрабатывается проверочным оператором. Прием применяется к условию задачи на описание, имеющему заголовок "или", так как именно в таких ситуаицях чаще всего и усматриваются тождественно истинные либо ложные фрагменты. Требуется, чтобы выражение b и первый антецедент не содержали неизвестных. Указатель "списокпосылок(1)" уточняет, что антецедент достаточно идентифицировать в посылках и не тратить время на рассмотрение условий. Уровень срабатывания равен 2.

Попытка усмотреть избыточность дизъюнктивного члена с неизвестными

Попытка усмотреть истинность либо ложность равенства нулю выражения с неизвестными может предприниматься в задаче на описание уже на уровне 1. Это делается, если равенство находится внутри дизъюнкции. Обычно дизъюнкции в списке условий задачи на описание возникают как следствие срабатывания обобщенного приема, учитывающего все возможные вырожденные случаи. Так как в действительности вырожденные случаи встречаются редко, то дизъюнкция при ближайшем рассмотрении обычно вырождается. Иногда такое рассмотрение обеспечивается уже нормализаторами приема, и тогда она вообще не попадает в список условий. Иногда она все же регистрируется в условиях, но при отбрасывании своих ложных членов быстро устраняется. Данный прием и предназначен для ускорения расчистки дизъюнкции. Его теорема имеет вид $\forall_a(\neg(a = 0) \to \neg(a = 0))$. Равенство $a = 0$ либо его отрицание является корневым операндом дизъюнктивного условия. Антецедент обрабатывается проверочным оператором при весьма малом лимите трудоемкости.

Упрощение равенства нулевому набору

Условие $\neg((x, y) = (0, 0))$ часто возникает в задачах, использующих координаты точек плоскости. Его нецелесообразно переформулировать в виде дизъюнкции отрицаний равенств нулю, так как эта дизъюнкция инициирует бесполезный разбор случаев. Поэтому нужны приемы, упрощающие указанное условие без его покоординатной расшифровки. Один из таких приемов заключается в вынесении наружу общего множителя координат x, y:

$$\forall_{abcde}\left(\neg\left(\left(\frac{ab}{c}, \frac{ad}{e}\right) = (0,0)\right) \leftrightarrow \neg\left(\left(\frac{b}{c}, \frac{d}{e}\right) = (0,0)\right) \ \& \ \neg(a = 0)\right)$$

Переменные b, c, d, e выделены указателем "единица"; указатель "заменазнака" относит возможные знаки "минус" к множителям числителя b, d. Прием срабатывает на уровне 1.

Проверочный оператор "усмне0"

Оператор предназначен для проверки истинности утверждений вида $\neg(a = 0)$. Обращение к нему имеет вид "усмне0(x1 x2 x3 x4)", где x1 - терм a; x2 - список посылок; x3 - набор комментариев. Если результат проверки позитивный, то выходной переменной x4 присваивается список использованных посылок. Как и во всех случаях обращений к пакетным операторам, данный список может оказаться существенно шире списка фактически использованных посылок. Список приемов оператора постоянно пополняется за счет понятий новых разделов. Впрочем, пока этот список сравнительно невелик. Чтобы проверочный оператор работал как можно быстрее, приходится вводить большое количество мелких приемов, ориентированных на легко обрабатываемые специальные случаи, и избегать приемов более общих, но зато и более трудоемких.

Перечислим основные приемы оператора:

1. Отбрасывание минуса.

   $\forall_a(a - \text{число} \ \& \ \neg(a = 0) \to \neg(-a = 0))$

   Антецеденты обрабатываются проверочными операторами. Указатель "спуск" блокирует альтернативные попытки, если применить прием не удалось. В этом случае сразу выдается отказ (т.е. выход из проверочного оператор по значению "ложь").

2. Отбрасывание показателя степени.

   $\forall_{ab}(a - \text{число} \ \& \ \neg(a = 0) \to \neg(a^b = 0))$

   Указатели такие же, как в предыдущем случае.

3. Отбрасывание модуля.

   Имеются два приема: для модуля вещественного и комплексного.

   $\forall_a(a - \text{число} \ \& \ \neg(a = 0) \to \neg(|a| = 0))$

   $\forall_a(a - \text{комплексное} \ \& \ \neg(a = 0) \to \neg(|a| = 0))$

   В первом случае используется логический символ "модуль", во втором - "Модуль". Заметим, что рассматриваемый проверочный оператор обслуживает сразу оба случая - вещественнозначный и комплекснозначный. Подмножество приемов, относящихся к операциям над комплексными числами, приводится ниже. Эти операции прорисовываются формульным редактором неотличимо от вещественнозначных. Соответствующие логические символы при этом разные. Как правило, название символа в комплекснозначном случае отличается от вещественнозначного заглавной первой буквой.

4. Исключение дроби.

   $$\forall_{ab}(b - \text{число} \ \& \ \neg(b = 0) \ \& \ a - \text{число} \ \& \ \neg(a = 0) \to \neg\left(\frac{a}{b} = 0\right))$$

   Все антецеденты обрабатываются проверочным оператором; имеется указатель "спуск".

5. Исключение произведения.

   $\forall_{ab}(\neg(a=0)\ \&\ \neg(b=0) \to \neg(ab=0))$

   Кроме указанных выше, прием также имеет указатель "дистрибразвертка(фикс (0 1 1))". Это позволяет обрабатывать произведения любого числа сомножителей: прием проверяет отличие от нуля каждого из них поочередно.

6. Попытка изменения знака у всех слагаемых.

   $\forall_{ab}(\neg(a-b=0) \to \neg(b-a=0))$

   Этот прием имеет две версии - вещественнозначную и комплекснозначную. В первом случае антецедент обрабатывается проверочным оператором, во втором - непосредственно извлекается из списка посылок. Выражение $-b$ идентифицируется с каким-то единственным слагаемым, перед которым стоит знак минус. Переменная a идентифицируется с остатком слагаемых. Для блокировки обратного перехода используется комментарий "перестановка". Обычно изменение знака бывает полезно для идентификации отрицания равенства с другим отрицанием, уже имеющимся в посылках.

7. Попытки усмотрения положительности либо отрицательности. Эти приемы оказываются результативными очень часто, так как позволяют воспользоваться мощным пакетом "усмменьше". Однако, они трудоемки, и уровень обращения к ним выбран равным 2 (проверка положительности) либо 3 (проверка отрицательности). Теоремы приемов таковы:

   $\forall_a(0 < a \to \neg(a=0))$

   $\forall_a(a < 0 \to \neg(a=0))$

   Антецедент обрабатывается проверочным оператором "усмменьше". Последний оператор анализирует утверждения вида $a < b$ и имеет два входных параметра a, b. Чтобы заблокировать повторы и заведомо ненужные попытки, введен ряд фильтров. Повторы блокируются комментарием (усмменьше 0 x1). Приемы не применяются при наличии комментария "перестановка" (см. предыдущий прием). Первый прием блокируется, если в посылках имеется строгое неравенство для a противоположного направления. Второй прием блокируется при рассмотрении заведомо неотрицательных величин (углы, расстояния, площади). Блокировка имеет место также для комплекснозначных выражений.

8. Использование строгого неравенства для произведения. Неравенства, устанавливающие строгий либо нестрогий знак произведения, упрощаются до тех пор, пока удается усмотреть некоторый строго положительный либо строго отрицательный сомножитель. Остаточное неравенство может все еще иметь произведение, и оно не преобразуется к виду дизъюнкции конъюнкций неравенств для сомножителей. Строгие неравенства такого вида можно использовать для усмотрения отличия сомножителя от нуля:

   $\forall_{ab}(0 < ab\ \&\ a - \text{число} \to \neg(a=0))$

   $\forall_{ab}(ab < 0\ \&\ a - \text{число} \to \neg(a=0))$

   Хотя прием нетрудоемок, имеются блокирующие фильтры: выражение a не должно быть константным; должен отсутствовать комментарий "перестановка".

9. Разность степеней с одинаковыми показателями. Если показатели степени - одинаковые рациональные числа с нечетными числителями, то их можно отбросить:

   $\forall_{abc}(c - \text{rational} \ \& \ \neg(\text{числитель}(c) - \text{even}) \ \& \ \neg(a - b = 0) \rightarrow \neg(a^c - b^c = 0))$

   Антецеденты обрабатываются проверочными операторами.

   Кроме того, в разделе имеется прием для отбрасывания степени положительного числа, из которого вычитается единица:

   $\forall_{ab}(\neg(1 - a = 0) \ \& \ 0 < a \ \& \ \neg(b = 0) \rightarrow \neg(a^b - 1 = 0))$

   Приемы для оставшихся аналогичных случаев не были востребованы при решении задач, и в разделе отсутствуют.

10. Отличие от нуля значения квадратного трехчлена с отрицательным дискриминантом.

    $\forall_{abcd}(b^2 - 4ac < 0 \rightarrow \neg(ad^2 + bd + c = 0))$

    Коэффициенты a, b, c суть десятичные записи. Поэтому антецедент можно проверять путем непосредственных вычислений, и он выделен указателем "программа".

11. Усмотрение из посылок специального вида. В разделе собраны около трех десятков приемов, использующих для усмотрения отличия значения выражения от нуля посылки различных специальных типов. Все они возникали для ускорения решения конкретных задач. Предполагается, что проверочный оператор применяется в ситуации, когда сопровождающие утверждения обеспечивают требования на о.д.з. Поэтому в теоремах его приемов антецеденты, относящиеся к о.д.з., обычно опускаются.

    $\forall_{abcd}(\neg(d(ab)^c + a = 0) \rightarrow \neg(a = 0))$

    $\forall_{abcd}(\neg(d(ab)^c - a = 0) \rightarrow \neg(a = 0))$

    $\forall_a(0 < |a| \rightarrow \neg(a = 0))$

    $\forall_{ab}(\neg(a = b) \rightarrow \neg(a - b = 0))$

    Здесь a - переменная. Имеются две версии этого приема: для вещественной и комплексной сумм.

    $\forall_a(\neg(\text{tg}a = 0) \rightarrow \neg(\text{sin}a = 0))$

    $\forall_a(\neg(\text{ctg}a = 0) \rightarrow \neg(\text{cos}a = 0))$

    $\forall_{abcd}(a^d b = c \ \& \ \neg(c = 0) \rightarrow \neg(a = 0))$

    $\forall_{ab}(a = b \ \& \ \neg(a = 0) \rightarrow \neg(b = 0))$

    Второй антецедент в последних двух приемах обрабатывается проверочным оператором.

    $$\forall_{abcdefgh}(\neg\left(\frac{ca}{d} + \frac{be}{f} = 0\right) \ \& \ cfg - hde = 0 \rightarrow \neg(ah + bg = 0))$$

    Второй антецедент выделен указателем "идентификатор". Коэффиценты g, h группируются из всех константных множителей. Прием усматривает в посылках условие отличия от нуля некоторой линейной комбинации выражений a, b

и проверяет, что текущая линейная комбинация этих выражений ей пропорциональна.

$\forall_{abc}(0 < a \ \& \ a < bc \to \neg(b = 0))$

Первый антецедент обрабатывается проверочным оператором.

$\forall_{a}(\neg(a - \text{even}) \ \& \ a - \text{целое} \to \neg(a = 0))$

Второй антецедент, по существу, избыточен, так как представляет собой о.д.з. для первого. Он обрабатывается проверочным оператором.

$$\forall_{abcde}\left(\frac{a^d b}{e} = c \ \& \ \neg(c = 0) \to \neg(a = 0)\right)$$

Второй антецедент обрабатывается проверочным оператором. Он имеет очень сильный ограничитель трудоемкости.

$\forall_{abc}(\neg(a - b = 0) \ \& \ 0 \le a \ \& \ 0 \le b \ \& \ c = b^2 \to \neg(a^2 - c = 0))$

Непосредственно в посылках берется первый антецедент. Выражения b, c суть десятичные записи. Второй и третий антецеденты обрабатываются проверочными операторами, а четвертый выделен указателем "идентификатор". Прием усматривает в посылках отрицание равенства разности нулю, представляющее собой результат "извлечения корня" из анализируемого отрицания равенства разности нулю.

$\forall_{abc}(\neg(a - b = 0) \ \& \ 0 \le a \ \& \ 0 \le b \ \& \ c = b^2 \to \neg(c - a^2 = 0))$

Прием аналогичен предыдущему.

$\forall_{abc}((\neg(a = 0) \ \vee \ \neg(b = 0) \ \vee \ \neg(c = 0)) \to \neg(a^2 + b^2 + c^2 = 0))$

$\forall_{ab}((\neg(a = 0) \ \vee \ \neg(b = 0)) \to \neg(a^2 + b^2 = 0))$

$\forall_{acd}(a^d = c \ \& \ \neg(c = 0) \to \neg(a = 0))$

Прием блокируется, если выражение c имеет a своим сомножителем.

$\forall_{ab}(b - \text{rational} \ \& \ \text{числитель}(b) - \text{even} \ \& \ \neg(a^b - 1 = 0) \to \neg(|a| - 1 = 0))$

Первые два антецедента обрабатываются проверочными операторами.

$\forall_{ab}(b - \text{rational} \ \& \ \text{числитель}(b) - \text{even} \ \& \ \neg(b = 0) \ \& \ \neg(|a| - 1 = 0) \to \neg(a^b - 1 = 0))$

$\forall_{abc}(\neg\left(a^{b/c} - 1 = 0\right) \ \& \ \neg(a = 0) \to \neg(b = 0))$

Второй антецедент обрабатывается проверочным оператором.

$\forall_{ab}(\neg(b = 0) \ \& \ |a| = b \to \neg(a = 0))$

Первый антецедент обрабатывается проверочным оператором.

$\forall_{ab}(\neg(a = 1) \to \neg(ab - b = 0))$

$\forall_{xaP}(x \in a \ \& \ a \subseteq \text{set}_y(P(y)) \ \& \ \neg(x = 0) \to \neg(x = 0))$

Смысл этого приема заключается в том, что для проверки отличия x от 0 привлекаются дополнительные посылки $P(x)$. Первые два антецедента берутся непосредственно из посылок. Так как x принадлежит множеству a, причем каждый элемент множества a обладает свойством P, то третьему антецеденту, обрабатываемому проверочным оператором, передается дополнительная посылка $P(x)$. Это обеспечивается указателем "занесениепосылки(3 значение(x40 x23))". Комментарий (содержится принадлежит) блокирует повторное применение данного приема.

$\forall_{ab}(\neg(a+1=0) \to \neg(a^b+1=0))$

$\forall_{ab}(\neg(a+1=0) \ \& \ a<0 \to \neg(a^b-1=0))$

Второй антецедент обрабатывается проверочным оператором.

$\forall_{ab}(\neg(a-b=0) \to \neg(a/b-1=0))$

$\forall_{abc}(\neg(\log_c(a/b)=0) \to \neg(a-b=0))$

$\forall_{ab}(\neg(a/b-1=0) \to \neg(a-b=0))$

Числитель и знаменатель дроби в последних двух приемах допускают перестановку.

$\forall_{ab}(\neg(\log_a b=0) \to \neg(b-1=0))$

$\forall_{abcn}(a^n b+c=0 \ \& \ \neg(c=0) \to \neg(a=0))$

Второй антецедент обрабатывается проверочным оператором.

12. Исключение знаменателя. Этот прием близко примыкает к предыдущей серии, однако вместо непосредственной идентификации в посылках обращается к проверочному оператору:

    $\forall_{ab}(\neg(a-b=0) \to \neg(a/b-1=0))$

13. Исключение логарифма.

    $\forall_{ab}(\neg(b-1=0) \to \neg(\log_a b=0))$

    Антецедент обрабатывается проверочным оператором.

14. Исключение синуса.

    $\forall_{ab}(\cos a=b \ \& \ \neg(1-|b|=0) \to \neg(\sin a=0))$

    Первый антецедент берется в списке посылок, второй обрабатывается проверочным оператором. Переменная b идентифицируется с константой.

$$\forall_{amn}(\cos\left(a+\frac{m\pi}{n}\right)=0 \ \& \ |m|<2n \to \neg(\sin a=0))$$

Первый антецедент берется в списке посылок, причем m, n идентифицируются, соответственно, с целочисленной и натуральной константой. Второй антецедент выделен указателем "программа".

$$\forall_{amn}(\sin\left(a+\frac{m\pi}{n}\right)=0 \ \& \ 0<|m| \ \& \ |m|<n \to \neg(\sin a=0))$$

Аналогично предыдущему.

$\forall_{abc}(\cos a=b \ \& \ a=2c \ \& \ \neg(b-1=0) \to \neg(\sin c=0))$

Первый антецедент берется в списке посылок. Второй выделен указателем "идентификатор", третий - обрабатывается проверочным оператором. Переменная b идентифицируется с константой.

$\forall_{abc}(\sin a=b \ \& \ a=2c \ \& \ \neg(b=0) \to \neg(\sin c=0))$

Аналогично предыдущему.

$$\forall_{nabcd}(n - \text{целое} \ \& \ \neg(a \cdot \text{нод}(b,c)|cd) \to \neg(\sin\left(\frac{\pi d}{a} + \frac{\pi b n}{c}\right) = 0))$$

Переменные a, b, c идентифицируются с натуральными константами, d - с целочисленной константой. Первый антецедент обрабатывается проверочным оператором, второй выделен указателем "программа".

$\forall_a(\operatorname{tg} a = b \ \& \ \neg(b = 0) \to \neg(\sin a = 0))$

Первый антецедент берется в списке посылок, второй обрабатывается проверочным оператором. Переменная b идентифицируется с константой.

$\forall_a(0 < a \ \& \ 0 < \pi - a \to \neg(\sin a = 0))$

Антецеденты обрабатываются проверочными операторами. Переменная a идентифицируется с неконстантным выражением. Прием применяется только в ситуациях, связанных с геометрией: либо имеется посылка, содержащая символ "угол", либо решается задача на описание, имеющая цель "стандравно". Такие цели возникают у вспомогательных задач, используемых для решения систем уравнений, полученных при анализе чертежа.

$\forall_a(0 < \operatorname{tg} a \to \neg(\sin a = 0))$

Антецедент берется в списке посылок.

$\forall_x(\neg\left(\frac{x}{\pi} - \text{целое}\right) \to \neg(\sin x = 0))$

Антецедент обрабатывается проверочным оператором. Проверяется, что выражение x имеет целочисленный параметр, а заголовок его отличен от символа "плюс".

$$\forall_{mnk}(m - \text{целое} \ \& \ n - \text{целое} \ \& \ \neg(n = 0) \ \& \ \neg(k - 1 = 0) \to \neg\left(\sin\frac{n\pi^k}{m} = 0\right))$$

Переменная k идентифицируется с десятичной записью. Первые три антецедента обрабатываются проверочным оператором, последний выделен указателем "программа".

$$\forall_{ABC}(\neg(\sin(\angle(ABC)) = 0) \to \neg\left(\sin\frac{\angle(ABC)}{2} = 0\right))$$

Антецедент берется в списке посылок.

15. Исключение косинуса.

$$\forall_{amn}(\sin\left(a + \frac{m\pi}{n}\right) = 0 \ \& \ |m| < 2n \to \neg(\cos a = 0))$$

Первый антецедент берется в списке посылок, причем m, n идентифицируются, соответственно, с целочисленной и натуральной константами. Второй антецедент выделен указателем "программа".

$\forall_{ab}(\sin a = b \ \& \ \neg(1 - |b| = 0) \to \neg(\cos a = 0))$

Первый антецедент берется в списке посылок, причем b идентифицируется с константным выражением. Второй антецедент обрабатывается проверочным оператором.

$\forall_{abc}(\cos a = b \ \& \ a = 2c \ \& \ \neg(b+1=0) \to \neg(\cos c = 0))$

$\forall_{abc}(\sin a = b \ \& \ a = 2c \ \& \ \neg(b=0) \to \neg(\cos c = 0))$

Аналогично предыдущему.

$$\forall_{amn}(\cos\left(a+\frac{m\pi}{n}\right) = 0 \ \& \ 0 < |m| \ \& \ |m| < n \to \neg(\cos a = 0))$$

Первый антецедент берется из списка посылок; второй и третий выделены указателем "программа". Переменные m, n идентифицируются с целочисленной и натуральной константами.

$\forall_{ab}(\sin a = b \ \& \ \neg(b=0) \to \neg(1+\cos a = 0))$

Первый антецедент берется из списка посылок, причем b идентифицируется с константным выражением.

$\forall_{ab}(\sin a = b \ \& \ \neg(1-2b^2=0) \to \neg(\cos(2a) = 0))$

$\forall_{ab}(\cos a = b \ \& \ \neg(2b^2-1=0) \to \neg(\cos(2a) = 0))$

Аналогично предыдущему.

$$\forall_{abcdn}(n - \text{целое} \ \& \ \neg(2a\cdot\text{нод}(b,c)|c(a-2d)) \to \neg(\cos\left(\frac{\pi d}{a}+\frac{\pi bn}{c}\right) = 0))$$

Первый антецедент обрабатывается проверочным оператором, второй выделен указателем "программа". Переменные a, b, c, d идентифицируются с целочисленными константами.

$$\forall_{abn}(n - \text{целое} \ \& \ \neg(2\text{нод}(a,b)|b) \to \neg\left(\cos\frac{\pi an}{b} = 0\right))$$

Аналогично предыдущему.

$\forall_{a}(\sin a = 0 \to \neg(\cos(na) = 0))$

Антецедент берется из списка посылок; переменная n идентифицируется с целочисленной константой.

$\forall_{abcd}(\operatorname{tg} a = b \ \& \ \operatorname{tg} c = d \ \& \ \neg(1-bd=0) \to \neg(\cos(a+c) = 0))$

Первые два антецедента берутся из списка посылок, третий - обрабатывается проверочным оператором. Переменные b, d идентифицируются с константными выражениями.

$\forall_{a}(0 < \pi + 2a \ \& \ 0 < \pi - 2a \to \neg(\cos a = 0))$

$\forall_{a}(0 < \pi/2 + a \ \& \ 0 < \pi/2 - a \to \neg(\cos a = 0))$

Антцеденты обрабатываются проверочными операторами. Переменная a идентифицируется с неконстантным выражением. Приемы срабатывают лишь в задачах с геометрическим контекстом (наличие углов и т.п.). Имеется версия второго приема с альтернативным условием на контекст: посылки должны содержать неравенство с $\pi/2$.

$\forall_{kn}((n$ – натуральное & k – целое & $0 \leq k$ & $0 < n - 2k \to \neg(\cos\frac{\pi k}{n} = 0))$

Антецеденты обрабатываются проверочными операторами, переменная n идентифицируется с неконстантным выражением.

$\forall_a(0 < a \;\&\; a < \pi/2 \to \neg(\cos a = 0))$

Оба антецедента берутся из списка посылок; переменная a идентифицируется с неконстантным выражением.

$\forall_{ABCa}(\angle(ABC) = \pi - a \;\&\; 0 < \pi/2 - \angle(ABC) \to \neg(\cos a = 0))$

Первый антецедент извлекается из списка посылок. Появление его в геометрических задачах достаточно вероятно. Второй антецедент обрабатывается проверочным оператором "усмменьше" и имеет вид, стандартный для группы приемов, усматривающих острые углы. Фильтр "контекст(посылка(x2)входит(угол x2))" ускоряющий. Он введен для того, чтобы уже на начальной стадии реализации приема отсечь негеометрические ситуации.

$\forall_{abxpq}(0 < ab \;\&\; 0 \leq x + p \;\&\; 0 \leq -x + q \;\&\; 0 < -2pa + \pi b \;\&\; 0 < \pi b - 2qa \to \neg(\cos(ax/b) = 0))$

Второй и третий антецедент, извлекаемые непосредственно из списка посылок, определяют диапазон возможных изменений переменной x. Знаки неравенств в них могут быть заменены на строгие. Первый, четвертый и пятый антецеденты обрабатываются проверочными операторами. Они устанавливают, что на указанном диапазоне изменения x аргумент косинуса изменяется от $-\pi/2$ до $\pi/2$.

$\forall_{abxk}(0 < 2ka - \pi \;\&\; 0 < 3\pi - 2ka \;\&\; a < x \;\&\; x < b \to \neg(\cos(kx) = 0))$

Прием аналогичен предыдущему. Из списка посылок извлекаются третий и четвертый антецеденты; в них допускаются нестрогие неравенства.

$\forall_a(0 < \operatorname{ctg} a \to \neg(\cos a = 0))$

Антецедент берется в спике посылок.

16. Исключение тангенса.

    $\forall_a(\neg(\sin a = 0) \to \neg(\operatorname{tg} a = 0))$

    Антецедент обрабатывается проверочным оператором. Прием сопровожден указателем "спуск", блокирующим альтернативные попытки проверочного оператора при неудаче.

17. Исключение котангенса.

    $\forall_a(\neg(\cos a = 0) \to \neg(\operatorname{ctg} a = 0))$

18. Линейная комбинация синуса и косинуса.

    $\forall_{ab}(\cos(2a) = b \;\&\; \neg(b = 0) \to \neg(\cos a - \sin a = 0))$

    Первый антецедент берется из списка посылок; b идентифицируется с константным выражением.

    $\forall_{ab}(\cos(2a) = b \;\&\; \neg(b = 0) \to \neg(\sin a - \cos a = 0))$

    $\forall_{ab}(\cos(2a) = b \;\&\; \neg(b = 0) \to \neg(\sin a + \cos a = 0))$

    Аналогично предыдущему.

$\forall_{abcde}(d = c/2 \ \& \ \operatorname{tg} d = e \ \& \ \neg(be^2 - 2ae - b = 0) \to \neg(a \sin c + b \cos c = 0))$

Второй антецедент берется в списке посылок. Переменные a, b, e идентифицируются с константными выражениями. Первый антецедент выделен указателем "идентификатор". Он проверяет, что аргумент тангенса является половиной аргумента синуса. Правая часть равенства обрабатывается нормализаторами "нормдробь", "стандплюс". Третий антецедент обрабатывается проверочным оператором.

$\forall_{abcd}(\operatorname{tg} c = d \ \& \ \neg(ad + b = 0) \to \neg(a \sin c + b \cos c = 0))$

Первый антецедент берется в списке посылок; переменные a, b, d идентифицируются с константными выражениями.

$$\forall_a(\sin\left(a + \frac{\pi}{4}\right) = 0 \to \neg(\sin a - \cos a = 0))$$

$$\forall_a(\sin\left(a + \frac{\pi}{4}\right) = 0 \to \neg(\cos a - \sin a = 0))$$

В последних двух приемах антецедент берется из списка посылок.

19. Вычитание синуса либо косинуса из единицы.

    $\forall_{ab}(\cos a = b \ \& \ \neg(b = 0) \to \neg(1 - \sin a = 0))$

    Первый антецедент извлекается из списка посылок, второй - обрабатывается проверочным оператором. Переменная b идентифицируется с константой.

    $\forall_{amn}(\cos(a + \frac{m\pi}{n}) = 0 \ \& \ 0 < |m| \ \& \ |m| < n \to \neg(1 - \sin a = 0))$

    Первый антецедент берется из списка посылок, остальные выделены указателем "программа". Переменные m, n идентифицируются с целой и натуральной константами.

    $\forall_a(\neg(\cos a = 0) \to \neg(1 - \sin a = 0))$

    Антецедент берется из списка посылок.

20. Линейный двучлен с синусом либо косинусом.

    $\forall_{abc}(0 < |a| - |b| \to \neg(a + b \sin c = 0))$

    $\forall_{abc}(0 < |a| - |b| \to \neg(a + b \cos c = 0))$

    Переменные a, b идентифицируются с константными выражениями. Антецедент обрабатывается проверочным оператором.

21. Косинус арктангенса.

    $\forall_a(\neg(\cos \operatorname{arctg} a = 0))$

22. Минус под косинусом.

    $\forall_a(\neg(\cos a = 0) \to \neg(\cos(-a) = 0))$

    Антецедент обрабатывается проверочным оператором. Указатель "спуск" блокирует альтернативные попытки.

23. Расстояние.

    $\forall_{AB}$(разныеточки$(A, B) \to \neg(l(AB) = 0))$

    Антецедент обрабатывается геометрическим проверочным оператором "разныеточки", устанавливающим различие двух точек. Чтобы предотвратить зацикливание, при обращении к оператору вводится комментарий "усмне0". Он заблокирует встречную попытку усмотреть различие точек при помощи оператора "усмне0".

    $\forall_{ABcdn}(l(AB) = cd^n\ \&$ разныеточки$(A, B) \to \neg(d = 0))$

    Первый антецедент берется в списке посылок. Указатель "единица" разрешает переменным c, n принимать вырожденные единичные значения. Второй антецедент обрабатывается проверочным оператором. Для отсечения негеометрических ситуаций введен ускоряющий фильтр "контекст(посылка(x5)входит(расстояние x5))". Он необходим из-за того, что консеквент теоремы приема не содержит каких-либо геометрических терминов.

24. Учет предельных условий.

    Задача может решаться в предположении о том, что значения некоторой переменной x берутся из сколь угодно малой проколотой окрестности (односторонней либо двусторонней) заданной точки a. Кванторы по размерам окрестности оказываются вынесены за скобку логического контекста задачи, однако на их наличие указывает фиктивная посылка "стремится($x\ a\ b$)". Здесь b - указатель типа окрестности (0 - двусторонняя, 1 - левая, 2 - правая). Эта посылка прорисовывается формульным редактором в виде $x \to a$. В случае односторонней окрестности справа от a дописывается $+0$ либо -0. В теоремах приемов общий вид посылки задается с помощью условного обозначения $x \to a\backslash b$.

    $\forall_{xabcd}(\neg(c = 0)\ \&\ (x \to a\backslash b) \to \neg(cx + d = 0))$

    Второй антецедент берется в списке посылок. Указатель "повтор(2)" разрешает использовать его многократно вдоль цепочки обращений пакетных операторов друг к другу. Первый антецедент обрабатывается проверочным оператором. Переменная x не входит в выражения c, d. Очевидно, при ненулевом c начиная с некоторого достаточно малого ε_0 в проколотой ε-окрестности точки a выражение $cx + d$ будет иметь ненулевое значение.

    $\forall_{xab}((x \to a\backslash b) \to \neg(x = 0))$

    Частный случай предыдущего приема, срабатывающий на меньшем уровне.

    $\forall_{xabcd}(\neg(d = 0)\ \&\ (x \to \infty) \to \neg(cx + d = 0))$

    $\forall_{xabcd}(\neg(d = 0)\ \&\ (x \to -\infty) \to \neg(cx + d = 0))$

    Отдельно разобранные случаи плюс-минус бесконечности.

    $\forall_{afn}((n \to \infty)\ \&\ \lim(f) = a\ \&\ \neg(a = 0) \to \neg(f(n) = 0))$

    Прием относится к числовым последовательностям. Посредством $\lim$ обозначен символ "пределпослед".

    $\forall_{abfz}(z -$ комплексное $\&\ (z \to a)\ \&\ \lim_{z \to a} f(z) = b\ \&\ \neg(b = 0) \to \neg(f(z) = 0))$

    Случай комплексного переменного. Первые два антецедента берутся из списка посылок. Третий антецедент выделен указателем "идентификатор". Он выполняет обращение к комплекснозначному нормализатору вычисления предела. Четвертый антецедент обрабатывается проверочным оператором. Указатель "отображение(x6)" означает, что $f(z)$ идентифицируется с произвольным выражением.

25. Условное выражение.

    $\forall_{abc}(\neg(a = 0)\ \&\ \neg(b = 0) \to \neg((a \text{ при } c,\ \text{ иначе } b) = 0))$

    Антецеденты обрабатываются проверочным оператором.

26. Усмотрение непрямого угла.

    Для усмотрения отличия величины угла $\angle(ABC)$ от $\pi/2$ в пакете создан несложный геометрический прием, основанный на следующей теореме:

    $\forall_{ABC}(\neg(l(AB)^2 + l(AC)^2 - l(BC)^2 = 0) \to \neg(\pi/2 - \angle(BAC)))$

    Он применяется, если в посылках имеются равенства, выражающие расстояния $l(AB), l(AC), l(BC)$ через известные параметры. Антецедент обрабатывается проверочным оператором. Описание приема содержит ряд чисто геометрических элементов, которые будут подробно рассмотрены в разделе книги, посвященном элементарной геометрии.

27. Гиперболические функции.

    Приведены всего два приема - для гиперболических косинуса и синуса:

    $\forall_a(\neg(\operatorname{ch} a = 0))$

    $\forall_a(\neg(a = 0) \to \neg(\operatorname{sh} a = 0))$

    Во втором случае антецедент обрабатывается проверочным оператором.

28. Использование дизъюнктивных посылок.

    Если имеется дизъюнктивная посылка, согласно которой значение выражения a либо не меньше некоторого значения b, либо не превосходит такого d, что $d < b$, то можно усмотреть отличие a от величины, находящейся внутри интервала (d, b). Это выполняется следующим приемом:

    $\forall_{abcdep}((p \leq a\ \&\ c\ \vee\ a \leq d\ \&\ e)\ \&\ b + p < 0\ \&\ d + p < 0 \to \neg(a + p = 0))$

    a идентифицируется с переменной; b, d, p - с константными выражениями. Разрешаются замены нестрогих неравенств на строгие. Второй и третий антецеденты обрабатываются проверочным оператором.

    Созданы два аналогичных приема для усмотрения отличия a от концевой точки интервала:

    $\forall_{abcdef}((f < a\ \&\ c\ \vee\ a \leq d\ \&\ e)\ \&\ d + b < 0\ \&\ f = -b \to \neg(a + b = 0))$

    $\forall_{abcdef}((a < f\ \&\ c\ \vee\ d \leq a\ \&\ e)\ \&\ 0 < b + d\ \&\ f = -b \to \neg(a + b = 0))$

    Неравенство для f строгое; неравенство для d может быть нестрогим либо строгим.

29. Физические величины.

    В физических задачах многое принимается "по умолчанию". Поэтому при проверке того, что физические характеристики реальных объектов (длина, масса, плотность, и т.п.) имеют ненулевые значения, достаточно бывает проверить отсутствие в контексте каких-либо намеков на вырожденную ситуацию. С точки зрения формальной логики, такой переход неаккуратен, однако при желании его можно отнести на счет принципа контекстной семантики, часто используемого и в других разделах решателя. Согласно этому принципу, логические контексты решателя иногда могут рассматриваться не буквально, а лишь как удобные для вычислений условные записи. Они должны восприниматься как фрагмент некоторого воображаемого внешнего контекста, который и будет давать полную аккуратную логическую расшифровку. С учетом сделанного замечания, приведем несколько типичных приемов усмотрения ненулевых значений физических величин.

$\forall_a(\neg(\text{масса}(a) = 0) \to \neg(\text{длина}(a) = 0))$

Уточняется, что выражение a не определяет векторную величину. Антецедент обрабатывается проверочным оператором.

$\forall_{abcd}(\text{извлечение}(a, b, c)\ \&\ \text{масса}(c) = d\ \&\ \neg(d = 0) \to \neg(\text{масса}(a) = 0))$

$\forall_{abcdx}(x = \text{масса}(a)\ \&\ \text{извлечение}(a, b, c)\ \&\ \text{масса}(c) = d\ \&\ \neg(d = 0) \to \neg(x = 0))$

Если из некоторого тела a извлечено вещество b, составившее тело c, причем масса последнего ненулевая, то и масса тела a ненулевая. Первые антецеденты берутся из списка посылок, последний - обрабатывается проверочным оператором.

$\forall_a(\neg(\text{плотность}(a) = 0))$

$\forall_{abcd}(\text{упругподвеска}(a, b, c)\ \&\ \text{движениетела}(c, d) \to \neg(\text{жесткость}(d) = 0))$

$\forall_A(\text{диск}(A)\ \&\ a = \text{радиус}(A) \to \neg(a = 0))$

30. Численные характеристики векторов.

$\forall_a(\neg(a = \text{вектор0} \to \neg(\text{скалумнож}(a, a) = 0))$

Антецедент обрабатывается проверочным оператором.

$\forall_{abc}(\neg(\text{компланарны}(a, b, c)) \to \neg(2\text{скалумнож}(a, b)\text{скалумнож}(a, c)\text{скалумнож}(b, c) + \text{скалумнож}(a, a)\text{скалумнож}(b, b)\text{скалумнож}(c, c) - \text{скалумнож}(a, b)^2\text{скалумнож}(c, c) - \text{скалумнож}(a, c)^2\text{скалумнож}(b, b) - \text{скалумнож}(b, c)^2\text{скалумнож}(a, a) = 0)$

$\forall_{ab}(\neg(\text{коллинеарны}(a, b)) \to \neg(\text{скалумнож}(a, a)\text{скалумнож}(b, b) - \text{скалумнож}(a, b)^2 = 0))$

Приемы усматривают отличие от нуля определителей Грама. В обоих случаях антецеденты обрабатываются проверочным оператором.

$\forall_{ABKabcd}(\text{прямкоорд}(K)\ \&\ \text{коорд}(A, K) = (a, b)\ \&\ \text{коорд}(B, K) = (c, d)\ \&\ \text{Вектор}(A)\ \&\ \text{Вектор}(B)\ \&\ \neg(\text{коллинеарны}(A, B)) \to \neg(ad - bc = 0))$

Прием усматривает отличие от нуля ориентированной площади невырожденного параллелограмма. Первые три антецедента берутся из списка посылок, последние три обрабатываются проверочными операторами.

$\forall_{ABCa}(\text{вектор}(AB) = a\text{вектор}(BC)\ \&\ \text{разныеточки}(A, C) \to \neg(a + 1 = 0))$

Первый антецедент берется из списка посылок, второй обрабатывается проверочным оператором. Прием имеет ускоряющий фильтр, проверяющий наличие посылки с операцией умножения вектора на число.

$\forall_{abc}(\neg(\text{компланарны}(a, b, c)) \to \neg(\text{длина}(a) = 0))$

$\forall_{KAabc}$(Вектор(A) & коорд$(A,K) = (a,b,c)$ & $\neg(A =$ вектор0$) \to \neg(a^2+b^2+c^2 = 0))$

В обоих приемах антецеденты берутся из списка посылок.

$\forall_a$(Вектор(a) & $\neg(a =$ вектор0$) \to \neg($длина$(a) = 0))$

Второй антецедент берется из списка посылок, первый - обрабатывается проверочным оператором.

$\forall_{abc}(\neg($компланарны$(a,b,c)) \to \neg($скалумнож$(a,$ вектумнож$(b,c)) = 0))$

Прием усматривает отличие от нуля смешанного произведения. Антцедент берется из списка посылок.

31. Комплексные числа.

Еще раз напоминаем, что комплекснозначные операции прорисовываются формульным редактором так же, как соответствующие им вещественнозначные, хотя фактически эти операции различны. Приводимый ниже перечень приемов, безусловно, требует значительного расширения. Однако, на текущий момент он достаточен для решения рассматривавшхся задач.

$\forall_{ab}(a -$ число & $b -$ число & $\neg(a = 0) \to \neg(a + bi = 0))$

$\forall_{ab}(a -$ число & $b -$ число & $\neg(b = 0) \to \neg(a + bi = 0))$

Антецеденты обрабатываются проверочными операторами.

$\forall_{ab}((\neg(a = 0) \ \vee \ \neg(b = 0)) \to \neg(a + bi = 0))$

Антецедент берется из списка посылок.

Приемы, исключающие минус, умножение, дробь и степень, аналогичны вещественнозначным аналогам, и мы их опускаем.

$\forall_z(\neg(z = 0) \to \neg($сопряженное$(z) = 0))$

Антецедент обрабатывается проверочным оператором.

$\forall_{zuv}(v =$ сопряженное(u) & $\neg(z + v = 0) \to \neg($сопряженное$(z) + u = 0))$

Первый антецедент выделен указателем "идентификатор". Его правая часть обрабатывается нормализатором "нормсопряженное", предпринимающим попытку избавиться от символа "сопряженное". Если это удается, то второй антецедент обрабатывается проверочным оператором.

$\forall_a(\neg(\cos a + i \sin a = 0))$

$\forall_a(\neg(-\cos a - i \sin a = 0))$

$\forall_{zab}(\mathrm{Im}(z) < b$ & $\mathrm{Im}(a) + b \leq 0 \to \neg(z + a = 0))$

Первый антецедент берется из списка посылок, второй - обрабатывается проверочным оператором.

$\forall_z(\neg(\mathrm{Re}(z) = 0) \to \neg(z = 0))$

Антецедент обрабатывается проверочным оператором. Выражение z неконстантное, причем для каждого его неконстантного слагаемого a посылки содержат равенство либо неравенство с корневым операндом $\mathrm{Re}(a)$. Прием имеет ускоряющий фильтр, проверяющий наличие посылки с заголовком "комплексное". Аналогичный прием для мнимой части имеет вид:

$\forall_z(\neg(\mathrm{Im}(z) = 0) \to \neg(z = 0))$

$\forall_{abz}(0 < a - |b| \,\&\, a < |z + b| \to \neg(z = 0))$

Первый антецедент обрабатывается проверочным оператором, второй берется из списка посылок.

$\forall_{za}(a < |z| \to \neg(z = 0))$

Антецедент берется из списка посылок. Выражение z неконстантное.

$\forall_{ab}(a - \text{число} \,\&\, b - \text{число} \,\&\, \neg(b = 0) \to \neg(\sin(a + bi) = 0))$

$\forall_{ab}(a - \text{число} \,\&\, b - \text{число} \,\&\, \neg(b = 0) \to \neg(\cos(a + bi) = 0))$

$\forall_{abc}(a - \text{число} \,\&\, b - \text{число} \,\&\, c - \text{число} \,\&\, \neg(a = 0) \,\&\, \neg(c = 0) \to \neg(\sin(a/(b+ci)) = 0))$

$\forall_{abc}(a - \text{число} \,\&\, b - \text{число} \,\&\, c - \text{число} \,\&\, \neg(c = 0) \to \neg(\cos(a/(b + ci)) = 0))$

Антецеденты обрабатываются проверочными операторами.

$\forall_{ab}(a - \text{число} \,\&\, b - \text{число} \,\&\, \neg(b = 0) \to \neg(\mathrm{tg}(a + bi) = 0))$

$\forall_{ab}(a - \text{число} \,\&\, b - \text{число} \,\&\, \neg(b = 0) \to \neg(\mathrm{ctg}(a + bi) = 0))$

32. Использование уравнений прямых и плоскостей.

В разделе находятся несколько приемов из аналитической геометрии.

Во-первых, имеется прием, усматривающий отличие от нуля определителя системы уравнений, составленной для нахождения точки пересечения двух различных пересекающихся прямых:

$\forall_{ABCKabcdef}(\text{коорд}(\text{прямая}(AB), K) = \mathrm{set}_{xy}(ax + by + c = 0 \,\&\, x - \text{число} \,\&\, y - \text{число}) \,\&\, \text{коорд}(\text{прямая}(AC), K) = \mathrm{set}_{uv}(du + ev + f = 0 \,\&\, u - \text{число} \,\&\, v - \text{число}) \,\&\, \text{разныепрямые}(\text{прямая}(AB), \text{прямая}(AC)) \,\&\, (p = ae - bd \,\vee\, p = bd - ae) \to \neg(p = 0))$

Первые два антецедента берутся из списка посылок, третий обрабатывается проверочным оператором, четвертый выделен указателем "идентификатор". На первый взгляд, прием выглядит странно. Он предлагает перебирать всевозможные пары уравнений прямых на плоскости, проходящих через общую точку, убеждаться в том, что они различны, вычислять определитель для коэффициентов уравнений (двумя способами - с учетом перемены знака), и лишь затем сравнивать найденный определитель с p. Эти действия чрезмерно трудоемки, и выполнение их "в общем случае" сильно замедлило бы процесс решения. Однако, все объясняется фильтром, проверяющим наличие комментария "разныепрямые". Такой комментарий вводится единственным приемом, решающим в задаче на исследование систему из двух линейных уравнений с двумя неизвестными. Он проверяет отличие от нуля определителя системы и, обращаясь к пакету "усмне0", вводит указанный комментарий. Учитывается сравнительно высокая вероятность происхождения уравнений системы из сопутствующих уравнений прямых.

Фильтр "не(равно(x27 x28))" отсекает вырожденный случай совпадения точек B, C. Заметим, что четвертый антецедент имеет заголовок "или". Выделение его указателем "идентификатор" означает, что дизъюнктивные члены будут обрабатываться последовательно, определяя таким образом перечисление возможных результатов идентификации. Окончательно будет отобран тот, для которого выполнение приема удастся довести до конца.

$\forall_{ABKabcdef}$(коорд(прямая$(AB), K) = \text{set}_{xyz}$(пропорцнаборы$((x+a, y+b, z+c), (d, e, f))$ & x – число & y – число & z – число) & $d = 0$ & $e = 0 \to \neg(f = 0))$

Если прямая в пространстве задается через свой направляющий вектор (d, e, f), то этот вектор не нулевой. Все антецеденты приема берутся в списке посылок. Очевидно, нужны еще хотя бы два аналогичных приема для первой и второй координат. Пока реализован лишь единственный случай, который фактически был востребован в задаче. Впрочем, такого рода недоделки типичны для многих разделов решателя, и их устранение может служить неплохим тренировочным материалом для обучения программированию на ГЕНОЛОГе.

$\forall_{ABCKabc}$(коорд(плоскость$(ABC), K) = \text{set}_{xyz}(ay + bz + c = 0$ & x – число & y – число & z – число) $\to \neg(a^2 + b^2 = 0))$

Прием аналогичен предыдущему: вектор нормали к плоскости ненулевой, причем снова рассмотрен лишь очень частный случай.

33. Исключение сигнума.

    $\forall_a(\neg(a = 0) \to \neg(\text{sg}a = 0))$

    Антецедент обрабатывается проверочным оператором.

34. Использование кванторной посылки.

    $\forall_{Aan}(A(n)$ & $\forall_i(A(i) \to \neg(a(i) = 0)) \to \neg(a(n) = 0))$

    Второй антецедент берется из списка посылок. Переменная a не выделена указателем "отображение", т.е. проверяется отличие от нуля значения некоторой функции a в точке n. Так как кванторный антецедент несет информацию об отличии от нуля значений данной функции, то естественно попробовать им воспользоваться, проверив истинность утверждения $A(n)$. Это делается с помощью вспомогательной задачи на доказательство.

    $\forall_{Aan}(A(n)$ & $\forall_i(A(i) \to a(i)$ – натуральное) $\to \neg(a(n) = 0))$

    Прием аналогичен предыдущему, но кванторная посылка берется более специального вида.

35. Целочисленные выражения.

    Следующие два приема усматривают ненулевое значение из соображений делимости.

    $\forall_{mnkp}(n$ – целое & $\neg(m|k)$ & $m|p$ & p – целое $\to \neg(mn + k + p = 0))$

    Переменные k, m идентифицируются с целочисленными константами. Второй антецедент выделен указателем "программа", прочие - обрабатываются проверочными операторами.

    $\forall_{mnk}(m$ – целое & n – целое & k – целое & $\neg(n - \text{even})$ & $\neg(k - \text{even}) \to \neg(mn + m + k = 0))$

    Все антецеденты обрабатываются проверочными операторами; переменные k, m идентифицируются с произвольными выражениями.

    $\forall_a(a$ – натуральное $\to \neg(a = 0))$

    Антецедент обрабатывается проверочным оператором.

    $\forall_{iknp}(\neg(k(\text{mod } n) = 0) \to \neg(p \cdot i(\text{mod } n) - pi + k = 0))$

Антецедент обрабатывается проверочным оператором.

$\forall_n(\neg(n! = 0))$

$\forall_{mn}(m -$ натуральное $\&\ n -$ натуральное $\&\ \neg(n-1=0) \to \neg(mn-1=0))$

Антецеденты обрабатываются проверочными операторами.

36. Мощность непустого множества.

    $\forall_A(\neg(A = \emptyset) \to \neg(\mathrm{card}A = 0))$

    Антецедент обрабатывается проверочным оператором.

37. Рациональные числа.

    $\forall_a(\neg(\text{знаменатель}(a) = 0))$

    $\forall_a(\neg(a - \mathrm{rational}) \to \neg(a = 0))$

    Антецедент берется из списка посылок.

    $\forall_a(a - \mathrm{rational}\ \&\ \neg(\text{числитель}(a) - \mathrm{even}) \to \neg(a = 0))$

    Второй антецедент берется из списка посылок, первый обрабатывается проверочным оператором.

38. Разное.

    В заключение приведем несколько приемов, понадобившихся в отдельных случаях для отдельных задач.

    $\forall_a(\neg(2a-1=0) \to \neg(\text{обрнормраспред}(a) = 0))$

    Прием относится к обратной функции нормального распределения. Антецедент обрабатывается проверочным оператором.

    $\forall_{ab}(\neg(b=0) \to \neg(a - \sqrt{b+a^2} = 0))$

    Часто встречающееся выражение с радикалом; антецедент обрабатывается проверочным оператором.

    $\forall_x(\neg(x=0) \to \neg(1 - \mathrm{exp}x = 0))$

    Антецедент обрабатывается проверочным оператором. Прием выделен указателем "спуск".

    $\forall_{mns}(m -$ целое $\&\ s -$ натуральное $\&\ n -$ целое $\&\ (\neg(m = 0)\ \vee\ \neg(n = 0)) \to \neg(m\pi^s + n = 0))$

    Антецеденты обрабатываются проверочными операторами. Указатель "группировка(x13)" обеспечивает группировку под общий коэффициент m всех слагаемых с π^s.

    $\forall_{fAmn}(\text{перестановка}(f, A)\ \&\ m -$ число $\&\ n -$ число $\&\ \neg(m-n=0) \to \neg(f(m) - f(n) = 0))$

    Первый антецедент берется в списке посылок, остальные обрабатываются проверочными операторами.

10.2 Приемы, связанные с десятичными записями

В разделе собраны приемы, выполняющие переход от десятичной дроби к простой несократимой. Все они связаны с символом "величина", так как в логическом представлении число $a_1 \dots a_n$ записывается термом "величина($a_1 \dots a_n$)". Здесь a_i - цифры, среди которых может иметься одна запятая. Напомним, что она представляет собой отдельный логический символ (,). Из-за ограничений, накладываемых прорисовкой термов, символ "величина" нельзя непосредственно использовать в теоремах приемов. Вместо него для этой цели введен вспомогательный символ "Величина", заменяемый компилятором на символ "величина".

Переход от десятичных дробей к простым представляет собой общую стандартизацию, которая может оказаться необходимой для срабатывания других приемов. Выделены несколько типичных случаев, когда эта стандартизация бывает полезна. Уточнение их происходит по мере надобности при обучении решателя.

Исключение десятичной дроби под степенью

Теорема приема прорисована текстовым редактором:

"длялюбого(x2 x3 если равно(x3 Величина(x2))то равно(Величина(x2)x3))"

Прием имеет заголовок "второйтерм". Антецедент выделен указателм "идентификатор". Он обеспечивает переход от десятичной дроби "Величина(x2)" к простой несократимой дроби x3. Для этого используется нормализатор "дробнаязапись", реализованный непосредственно на ЛОСе.

Прием проверяет, что десятичная запись содержит символ запятой. Проверяется, что эта запись либо является показателем степени и имеет не более 4 цифр, либо является основанием степени, показатель которой содержит неизвестную текущей задачи на описание. При этом проверяется, что текущее условие имеет еще хотя бы одну степень с неизвестным показателем и численным основанием.

Если преобразуется условие задачи на описание, причем показатель степени содержит неизвестные, то проверяется наличие неизвестной, входящей в это условие вне степенного выражения, имеющего то же самое численное основание. Очевидно, в противном случае дополнительная стандартизация численных оснований степени не нужна.

Прием имеет указатель "нормализатор", обеспечивающий необходимую простейшую стандартизацию внешнего контекста заменяемой десятичной записи (устранение многоэтажных дробей и т.п.). Уровень срабатывания равен 0.

Исключение десятичной дроби под тригонометрической операцией

Теорема приема такая же, как выше. Прием отличается лишь условиями срабатывания: применяется всякий раз, как только десятичная дробь оказывается расположена внутри терма, имеющего своим заголовком один из символов "синус", "косинус", "тангенс", "котангенс". При этом не должно быть внешнего терма с заголовком "арксинус", "арккосинус" либо "арктангенс".

Исключение десятичной дроби - коэффициента знаменателя

Теорема приема такова:

"длялюбого(x1 x2 x3 x4 x5 если равно(Величина(x2)x6)равно(x6 дробь(x1 x3))то равно(дробь(x4 умножение(x5 Величина(x2)))дробь(умножение(x3 x4)умножение(x1 x5))))"

Отличие от предыдущей версии теоремы заключается в том, что сразу находятся числитель x1 и знаменатель x3 простой числовой дроби, и происходит необходимая их перегруппировка между числителем и знаменателем внешнего дробного выражения. Преобразование оказывается ненужным в областях, где используются приближенные вычисления с десятичными дробями (физика, химия, и т.п.). Поэтому введен фильтр, блокирующий его выполнение при наличии в задаче понятий, относящихся к таким областям.

10.3 Приемы символа "число"

Утверждение "число(x)" означает, что x есть вещественное число. Начнем с перечисления приемов справочников, созданных для символа "число".

Справочник "родобъекта" указывает, что этот символ является названием одного из основных типов объектов. Справочник "род" определяет, что единственным надтипом типа "число" является "комплексное". Напомним, что различие двух основных подтипов одного и того же типа либо двух "корневых" основных типов влечет различие соответствующих объектов. Справочники "арность" и "предикатныйсимвол" указывают, что символ "число" является обозначением одноместного предиката. Справочник "повтор" сообщает, что имеется нормализатор "повторчисло", используемый для получения повторяющихся вхождений числовых подвыражений. Справочник "числовойатом" ссылается на оператор "числовойатом(...)", перечисляющий все атомарные неконстантные числовые подвыражения заданного терма. Атомарным числовым подвыражением считается либо числовая переменная, либо вещественнозначная операция, имеющая хотя бы один нечисловой операнд.

Кроме утверждения "число(x)", введено утверждение "Число(x)", означающее, что x - либо вещественное число, либо один из символов плюс-минус бесконечности. Для него созданы приемы справочников формульного редактора, однако в процессах решения задач данное утверждение пока не используется.

Приемы подбора примера

В разделе собраны несколько приемов, обеспечивающих подбор примера числа, удовлетворяющего условиям простейших типов.

$\forall_a(a = 0 \rightarrow a - \text{число})$

Прием имеет заголовок "подборзначений". Он срабатывает на уровне 0 в задачах на описание, имеющих цель "пример". a - неизвестная задачи, входящая в единственное условие "число(a)". Для неизвестной выбирается значение 0. Указатель "новый" обеспечивает слежение за исключением прочих содержащих a условий. После того, как все эти условия, кроме "число(a)", окажутся исключены, вес последнего условия, скорее всего, будет значительно больше нуля. Однако, как только текущий уровень N станет на единицу меньше данного веса, будет предпринят цикл повторных сканирований условия "число(x)", с уровнями от 0 до N, и прием сработает. Заметим, что данный прием часто применяется в связанных с числами задачах "теоретического" типа.

Как уже говорилось ранее, прием "подборзначений" реализует попытку решения задачи путем сведения ее к вспомогательной задаче. Условия последней получаются при замене исходных условий, идентифицированных с членами консеквента теоремы приема, на антецеденты, выделенные указателем "подборзначений". В нашем случае условие "число(x)" будет заменено на $x = 0$.

$\forall_x (x = 1 \to \neg(x = 0))$

Прием аналогичен предыдущему. Он срабатывает на уровне 1. Кроме условия $\neg(x = 0)$, допускается наличие условия "число(x)" либо "комплексное(x)".

$\forall_{ax} (a -$ число $\& \; x = a + 1 \to \neg(x = a))$

Прием является обобщением предыдущего. Первый антецедент обрабатывается проверочным оператором, второй - выделен указателем "подборзначений".

$\forall_{abx} (x -$ число $\& \; a -$ число $\& \; 0 < a \; \& \; x = 0 \to \neg(a + bx = 0))$

Прием срабатывает в задачах на описание, имеющих цель "пример". x - неизвестная задачи, входящая в выражение a. Прием пытается упростить левую часть равенства нулю, подставив вместо x ноль. Проверяется отсутствие содержащих x условий, не имеющих вида "число(x)". Первые три антецедента обрабатываются проверочными операторами, четвертый - выделен указателем "подборзначений".

$\forall_{ab} (a -$ число $\& \; b -$ число $\& \; 0 < a \; \& \; 0 < b \to \neg(a + b = 0))$

Выражения a, b содержат неизвестные. Прием предпринимает попытку обеспечить ненулевое значение суммы за счет положительных значений слагаемых. Предварительно проверяется, что не усматривается неположительность слагаемых a, b. Первые два антецедента обрабатываются проверочными операторами, следующие два - выделены указателем "подборзначений". Прием срабатывает на уровне 4. Заметим, что при его применении не накладываются ограничения на прочие условия с неизвестными.

Усмотрение вещественнозначного выражения

Утверждение "число($f(t_1 \dots t_n)$)", для которого справочник "тип" усматривает числовой тип значений операции f, заменяется на логическюу константу "истина". Теорема приема - "родобъекта(число)", заголовок приема - "родобъекта". Прием срабатывает на уровне 0.

Усмотрение противоречивых указаний на тип объекта

Та же самая теорема приема "родобъекта(число)" порождает еще два приема. Первый из них имеет заголовок "различимы". Он заменяет на константу "ложь" утверждение вида "число(x)", если в контексте встречается такое утверждение $P(x)$, что P - один из основных типов объектов, не являющийся ни надтипом, ни подтипом типа "число". Второй прием имеет заголовок "смзначение". Он заменяет на константу "ложь" утверждение "число($f(t_1 \dots t_n)$)", если справочник "тип" усматривает нечисловой тип значений операции f.

Обращение к проверочному оператору "усмчисло"

Иногда числовой тип значений выражения явно не указан, однако может быть усмотрен с помощью несложных логических переходов. Для такого усмотрения введен проверочный оператор "усмчисло". Его приемы будут рассмотрены ниже; здесь же укажем прием сканирования задачи, обращающийся к данному оператору. Теорема приема имеет вид $\forall_a(a - \text{число} \to a - \text{число})$; заголовок приема - "второйтерм". Антецедент обрабатывается проверочным оператором. Уровень срабатывания приема равен 2.

Хотя с логической точки зрения преобразование приема тривиально, он требует крайне аккуратного управления. Утверждение "число(x)" во многих случаях нужно сохранять в контексте, даже если оно избыточно. Например, должны сохраняться условия "число(x)" задач на описание, характеризующие числовые переменные x. При отсутствии этих условий будут заблокированы многие необходимые для решения приемы. Поэтому прием имеет многочисленные фильтры. Приведем некоторые из ситуаций, в которых применение его к утверждению "число(A)" заблокировано:

1. Утверждение находится в посылках.
2. Утверждение является условием задачи на описание, причем A - переменная.
3. Утверждение является антецедентом теоремы, возникающей при логическом выводе в базе теорем; A - переменная.
4. Утверждение используется в условии задачи на описание, выделенном комментарием "серия". Это условие является параметрическим описанием, причем A - параметр данного описания.
5. Утверждение используется в параметрическом описании класса (кроме случаев задач на описание, имеющих цель "программа").

Усмотрение ответа задачи на описание

Если остается единственное содержащее неизвестную условие "число(x)" задачи на описание, то ответ выдается рассмотренным ранее приемом, реализованным на ЛОСе. Этот прием срабатывает, если единственное содержащее неизвестную условие определяет тип ее значения. На ГЕНОЛОГе реализованы еще несколько приемов, усматривающих ответ при наличии слабых дополнительных ограничений, сопровождающих условие "число(x)".

Если дополнительное ограничение состоит в непринадлежности числа известному множеству, то применяется следующий прием:

$a - \text{число} \ \& \ \neg(a \in \{;b\})$

Он имеет заголовок "ответзадачи". Теорема приема - конъюнкция утверждений, идентифицируемых со всеми содержащими неизвестные условиями задачи. Переменная a идентифицируется с неизвестной, переменная b - с выражением без неизвестных. Указатель "смответ(2)" определяет выбор точки привязки во втором конъюнктивном члене теоремы приема.

Если дополнительных ограничений несколько, причем заключаются они в отличии числа от некоторых заданных чисел - отдельных либо образующих серии, - то применяется прием такого вида:

$a - \text{число} \ \& \ \forall_n(n - \text{целое} \ \& \ f(n) \to \neg(a = g(n))) \ \& \ \neg(a = b)$

Второй член конъюнкции - кванторная импликация, определяющая условие отличия числа a от каждого элемента числовой серии $g(n)$. Третий член - условие отличия числа от заданного известного значения b. Указатель "серия(2 3)" разрешает наличие в списке условий задачи любого количества утверждений, идентифицируемых со вторым либо третьим членами. Указатель "смответ(2)" определяет выбор точки привязки во втором члене, который таким образом должен идентифицироваться хотя бы с одним условием. Условий, идентифицируемых с третьим членом, может не быть вовсе.

Наконец, предусмотрен прием для случая, когда дополнительное условие заключается в иррациональности числа:

a – число & $\neg(a$ – rational$)$

Прием имеет указатель "смответ(2)".

Исключение избыточного указания на числовую переменную

В дополнение к уже упомянутым выше возможностям исключения избыточного утверждения "число(x)", приводятся следующие приемы:

$\forall_a(a$ – целое $\rightarrow a$ – число$)$

$\forall_a(a$ – натуральное $\rightarrow a$ – число$)$

Антецедент берется из контекста. Таким образом, имеется более сильная характеризация типа числовой переменной, и утверждение "число(a)" оказывается не нужным. Указатель "сопровождение" разрешает применение приема даже в случаях, когда преобразуемое утверждение используется для сопровождения по о.д.з. Прием имеет сразу два уровня срабатывания - 1 и 3, так как к моменту повторной попытки в контексте уже может появиться искомый антецедент.

$\forall_{xa}(x = a$ & a – число $\rightarrow x$ – число$)$

Прием применяется в условии задачи на описание, имеющей цель "пример". Он исключает указатель типа неизвестной после того, как найдено ее значение. x - неизвестная, a - выражение без неизвестных. Первый антецедент идентифицируется с условием задачи, второй - обрабатывается проверочным оператором. Уровень срабатывания равен 0.

$\forall_{amn}(\lambda_k(a(k), k$ – число & $k \in \{m, \ldots, n\}) = \lambda_k(a(k), k \in \{m, \ldots, n\}))$

Утверждение "число(k)" оказывается здесь избыточным, так как принадлежность k конечному отрезку целых чисел уже гарантирует числовой тип значений. Переменная a выделена указателем "отображение".

Усмотрение истинности дизъюнкции с двумя отрицаниями числовых равенств

$\forall_{abcx}(a$ – число & b – число & $\neg(a - b = 0) \rightarrow \neg(x = a)\ \vee\ \neg(x = b)\ \vee\ c)$

Появление отрицаний равенства одного и того же объекта x объектам a, b в дизъюнкции означает, что дизъюнкция истинна всякий раз, как только объекты a, b различны. Антецеденты обрабатываются проверочными операторами. Уровень срабатывания приема равен 2.

Устранение квантора

Раздел содержит несколько простых приемов, усматривающих тождественную истинность либо ложность кванторного утверждения, либо сводящих это утверждение к бескванторному. Во всех случаях под квантором имеется вхождение символа "число".

$\exists_a(a - \text{число})$

Прием имеет заголовок "второйтерм" и заменяет квантор на константу "истина".

$\neg(\forall_a(\neg(a - \text{число})))$

Аналогично предыдущему.

$\forall_{fg}(\neg(\exists_x(f(x) \ \& \ x - \text{число} \ \& \ \neg g(x))) \rightarrow \forall_x(f(x) \ \& \ x - \text{число} \rightarrow g(x)))$

Прием применяется при редактировании ответа задачи на описание. Если ответ содержит кванторную импликацию без неизвестных, некоторый антецедент которой имеет вид "число(x)", то предпринимается попытка установить тождественную истинность импликации. Для этого рассматривается ее отрицание, вынесенное в антецедент теоремы приема, и подкванторные утверждения квантора существования явно разрешаются относительно x. Здесь используется нормализатор, обращающийся к вспомогательной задаче на описание. Затем предпринимается попытка усмотреть истинность антецедента с помощью задачи на доказательство. Прием имеет заголовок "второйтерм" и срабатывает на уровне 4.

$\forall_c(\exists_{de}(d - \text{число} \ \& \ e - \text{число} \ \& \ \neg(d = 0) \ \& \ \neg(e = 0) \ \& \ c = de) \leftrightarrow \neg(c = 0) \ \& \ c - \text{число})$

Существование двух ненулевых чисел, произведение которых равно данному числу, эквивалентно отличию последнего от нуля.

$\forall_a(\exists_{bc}(b - \text{rational} \ \& \ c - \text{rational} \ \& \ \neg(\text{знаменатель}(b) - \text{even}) \ \& \ \neg(\text{знаменатель}(c) - \text{even}) \ \& \ a = bc \ \& \ b - \text{число} \ \& \ c - \text{число}) \leftrightarrow a - \text{rational} \ \& \ \neg(\text{знаменатель}(a) - \text{even}) \ \& \ a - \text{число})$

Существование двух рациональных чисел с нечетными знаменателями, произведение которых равно данному числу, эквивалентно тому, что последнее само является рациональным и имеет нечетный знаменатель.

$\forall_a(\neg(\forall_x(x - \text{число} \rightarrow x = a)))$

Прием усматривает, что не каждое число равно a. Он имеет заголовок "второйтерм".

$\forall_{an}(\exists_x(x - \text{число} \ \& \ \forall_i(i \in \{1, \ldots, n\} \rightarrow \neg(x = a(i)))))$

Прием усматривает существование числа, отличного от чисел заданного списка. Указатель "развертка(фикс(0 2 2))" определяет идентификацию кванторной импликации с конъюнкцией отрицаний равенств $\neg(x = a(1)), \ldots, \neg(x = a(n))$.

Перенесение в антецедент указателя числовой переменной

Кванторная импликация обычно приводится к такому виду, чтобы утверждения, накладывающие слабые ограничения на контекст, размещались в антецедентах. Например, в антецеденты переносятся указатели типа значения переменной. Применительно к типу "число" такое перенесение выполняется следующим приемом:

$\forall_{AB}(\forall_{xy}(A(x, y) \ \& \ B(x, y) \rightarrow \neg(x - \text{число})) \leftrightarrow \forall_{xy}(x - \text{число} \ \& \ A(x, y) \rightarrow \neg B(x, y)))$

Указатель "кортежпеременных(х24)" показывает, что y идентифицируется с произвольным (возможно, пустым) остатком переменных кванторной приставки, из которой выделена переменная x. Указатель "внешнийквантор(фикс(0 1))" отменяет попытку идентификации с преобразованием квантора существования в отрицание кванторной импликации. Заметим, что теорема приема имеет в антецедентах две функциональных переменных - $A(x, y)$ и $B(x, y)$. Чтобы уточнить принцип их идентификации, введен указатель "антецедент(х27 фикс(0 1))". Он означает идентификацию $B(x, y)$ с каким-то отдельным антецедентом, и тогда $A(x, y)$ будет идентифицировано с остатком. Имеется фильтр, проверяющий, что утверждение $B(x, y)$ не имеет заголовка "число" и не является неравенством.

Символы бесконечности

Следующие приемы эквивалентной замены усматривают, что символы бесконечности и вспомогательный символ "неопред" (иногда используемый для указания на то, что значение не определено) не являются числами. Все эти приемы имеют заголовок "второйтерм".

$\neg(\infty - \text{число})$

$\neg(-\infty - \text{число})$

$\neg(\text{неопред} - \text{число})$

Подбор примера для отрицания числового равенства

Если условие задачи на описание, имеющей цель "пример", представляет собой отрицание числового равенства, то предпринимается попытка явного разрешения этого равенства относительно одной из неизвестных:

$\forall_{AB}((A = 0) = B \to \neg(A = 0) \leftrightarrow \neg B)$

Антецедент имеет обращение к нормализатору, разрешающему уравнение $A = 0$ относительно некоторой неизвестной х1, имеющей единственное вхождение в A. Она идентифицируется с помощью указателя "контекст(неизвестная(х1)входит(х1 х26))". Уровень срабатывания приема равен 5.

Усмотрение существования значения неизвестной

В разделе собраны приемы, позволяющие отбрасывать несущественные неизвестные задачи на описание, если усматривается существование требуемых их значений. Все эти приемы имеют заголовок "связка".

$\forall_{an}(\exists_x(x - \text{число} \ \& \ \forall_i(i \in \{1, \ldots, n\} \to \neg(x = a(i)))))$

Неизвестная x встречается только в условии "число(x)" и условиях, выражающих отличие ее от заданных значений $a(i)$. Последние определяются без участия x. Кванторная импликация выделена указателем "развертка".

$\forall_{an}(\exists_x(x - \text{число} \ \& \ x < p \ \& \ \forall_i(i \in \{1, \ldots, n\} \to \neg(x = a(i)))))$

Аналогично предыдущему, но дополнительно x удовлетворяет некоторому неравенству. Противоположная часть этого неравенства от x не зависит. Указатель "вариант(фикс(0 2 2)меньшеилиравно)" разрешает нестрогий знак неравенства; указатель "дробь(фикс(0 2 2))" разрешает перестановку частей неравенства.

$\forall_{abckn}(\neg(a = 0)\ \&\ 0 < k \to \exists_x(\forall_i(i \in \{1, \ldots, n\} \to \neg(x = c(i)))\ \&\ x -$ число $\&\neg(ax^k + b = 0)))$

Еще один специальный случай: существует значение неизвестной x, отличное от величин $a(i)$ и не обращающее в нуль выражение $ax^k + b$. Антцеденты, проверяющие положительность k и отличие от нуля a, обрабатываются проверочными операторами. Указатель "вариант(фикс(0 2 2)целое натуральное рациональное)" расширяет список допустимых заголовков идентифицируемого утверждения "число(x)".

Очевидно, список приемов подобного типа мог бы быть значительно расширен. Читателю предлагается самостоятельно развить более общий аппарат исключения несущественных числовых неизвестных.

Проверочный оператор "усмчисло"

Оператор имеет следующие приемы:

1. Посылка, указывающая числовой подтип.

   $\forall_a(a -$ rational $\to a -$ число$)$

   $\forall_a(a -$ целое $\to a -$ число$)$

   $\forall_a(a -$ натуральное $\to a -$ число$)$

   Антецеденты берутся из списка посылок.

2. Использование тождества в посылках.

   $\forall_{ab}(a -$ число $\&\ b = a \to b -$ число$)$

   Первый антецедент обрабатывается проверочным оператором, второй - идентифицируется с тождеством из посылок.

3. Мощность конечного множества.

   $\forall_a($конечное$(a) \to \mathrm{card}(a) -$ число$)$

   Антецедент обрабатывается проверочным оператором.

4. Условное выражение.

   $\forall_{abc}(a -$ число $\&\ b -$ число $\to (a$ при $c,$ иначе $b) -$ число$)$

   Антецеденты обрабатываются проверочными операторами.

5. Принадлежность конечному числовому списку.

   $\forall_{an}(n \in \{a;\}\ \&\ \{a;\} \subseteq \mathbf{R} \to n -$ число$)$

   Первый антецедент берется из списка посылок, второй - обрабатывается проверочным оператором.

6. Принадлежность числовому промежутку.

   $\forall_{abcde}(a \in [b, c] \to a -$ число$)$

   Прием допускает промежутки с концами любого типа. Антецедент берется из списка посылок.

7. Принадлежность отрезку целых чисел.

   $\forall_{amn}(a \in \{m, \ldots, n\} \to a -$ число$)$

8. Комплекснозначная операция над вещественными числами.

   $\forall_{ab}(a - \text{число}\ \&\ b - \text{число} \to a + b - \text{число})$

   $\forall_{ab}(a - \text{число}\ \&\ b - \text{число} \to ab - \text{число})$

   $\forall_{a}(a - \text{число} \to -a - \text{число})$

   $\forall_{ab}(a - \text{число}\ \&\ b - \text{число} \to a/b - \text{число})$

   $\forall_{ab}(a - \text{число}\ \&\ b - \text{число}\ \&\ 0 < a \to a^b - \text{число})$

   $\forall_{ab}(a - \text{число}\ \&\ b - \text{целое} \to a^b - \text{число})$

   Антецеденты обрабатываются проверочными операторами. Все рассмотренные здесь операции имеют комплекснозначные версии (их названия начинаются с большой буквы).

9. Координата точки либо вектора.

   $\forall_{ABcd}(\text{коорд}(A, B) = (c, d) \to c - \text{число})$

   $\forall_{ABcd}(\text{коорд}(A, B) = (c, d, e) \to c - \text{число})$

   Антецедент берется из списка посылок; кроме приведенных двух приемов, имеются еще три, соответствующие перестановкам координат.

10. Использование включения в множество вещественных чисел.

    $\forall_{ab}(a \in b\ \&\ b \subseteq \mathbf{R} \to a - \text{число})$

    Оба антецедента берутся из списка посылок.

11. Нижняя либо верхняя грань.

    $\forall_{ab}(a \subseteq \mathbf{R}\ \&\ \text{нижняягрань}(b, a) \to b - \text{число})$

    $\forall_{ab}(a \subseteq \mathbf{R}\ \&\ \text{верхняягрань}(b, a) \to b - \text{число})$

    Первый антецедент обрабатывается проверочным оператором, второй берется из списка посылок.

12. Наименьший либо наибольший элемент.

    $\forall_{cd}(c - \text{set}\ \&\ c \subseteq \mathbf{R}\ \&\ \text{наименьший}(d, c) \to d - \text{число})$

    $\forall_{cd}(c - \text{set}\ \&\ c \subseteq \mathbf{R}\ \&\ \text{наибольший}(d, c) \to d - \text{число})$

    Первые два антецедента обрабатываются проверочным оператором.

13. Дифференцируемость в точке.

    Условие существования производной в заданной точке кодируется как утверждение о численном значении выражения для производной. Если в посылках уже имеется такое утверждение, то для усмотрения аналогичного утверждения, отличающегося лишь переобозначением связанной переменной, используется следующий прием:

    "длялюбого(x2 x3 если число(производная(отображение(x1 число(x1)значение(x2 x1))x3))то число(производная(отображение(x4 число(x4)значение(x2 x4))x3)))".

    Заметим, что при формульной прорисовке связанные переменные x1, x4 оказываются спрятаны, и теорема имеет мало информативный вид:

    $$\forall_{bc}(\frac{db(c)}{dc} - \text{число} \to \frac{db(c)}{dc} - \text{число})$$

14. Значение случайной величины.

    $\forall_{fAb}$(случвеличина$(f, A) \to f(b)$ – число)

    Антецедент берется из списка посылок.

15. Предел сходящейся числовой последовательности.

    $\forall_{ab}$(сходится(a) & $\lim(a) = b$ & последовательность$(a, \mathbf{R}) \to b$ – число)

    Первые два антецедента берутся из списка посылок, третий обрабатывается проверочным оператором.

16. Элемент числовой последовательности.

    $\forall_{fn}$(последовательность$(f, \mathbf{R})$ & n – натуральное $\to f(n)$ – число)

    Первый антецедент берется из списка посылок, второй - обрабатывается проверочным оператором.

17. Использование параметрического описания.

    $\forall_{fA}(f(y)$ – число & $\exists_y(x = f(y)$ & $A(y)) \to x$ – число)

    Второй антецедент берется из списка посылок и представляет собой параметрическое описание значений, принимаемых выражением x. Первый антецедент обрабатывается проверочным оператором, которому придается дополнительная посылка $A(y)$.

18. Значение числовой функции.

    $\forall_{fi}(\mathrm{Val}(f) \subseteq \mathbf{R} \to f(i)$ – число)

    Антецедент обрабатывается проверочным оператором. Переменная f не выделена указателем "отображение", т.е. выражение $f(i)$ идентифицируется с термом "значение(...)".

    $\forall_{fABx}(x \in A$ & Отображение(f, A, B) & $B \subseteq \mathbf{R} \to f(x)$ – число)

    Первый и третий антецеденты обрабатываются проверочными операторами, второй - берется из списка посылок.

Нормализатор числовых равенств "нормчисло"

Простейшая стандартизация новых числовых равенств, создаваемых приемами замены либо вывода, выполняется нормализатором "нормчисло". К уравнениям он обычно не применяется. Нормализатор имеет следующие приемы:

1. Усмотрение ложности равенства при помощи оператора "усмне0".

   $\forall_{ab}(\neg(a - b = 0) \to \neg(a = b))$

   Антецедент обрабатывается проверочным оператором. Достаточно сильный ограничитель трудоемкости делает попытку усмотрения быстрой. Указатель "коммутативно(фикс(0 1))" отменяет ненужную попытку перестановки операндов a, b, удваивая таким образом эффективность приема. Уровень срабатывания приема в пакете равен 3.

2. Сокращение обеих частей равенства на общий ненулевой множитель.

   $\forall_{abc}(\neg(a = 0) \mathbin{\&} b - \text{число} \mathbin{\&} c - \text{число} \to (ab = ac \leftrightarrow b = c))$

   Антецеденты обрабатываются проверочными операторами. Переменные b, c выделены указателями "единица" и "заменазнака". Чтобы можно было сокращать равенство в случаях, когда отличие от нуля общего множителя не усматривается, введен еще один прием:

   $\forall_{abc}(b - \text{число} \mathbin{\&} c - \text{число} \to ab = ac \leftrightarrow a = 0 \ \vee \ b = c)$

   Этот прием активируется при наличии комментария "нормуравн". Иначе - применяется первый прием. Заменяющий терм второго приема обрабатывается рядом нормализаторов. Прежде всего, к равенству $a = 0$ применяются нормализаторы "нормусм" и "нормчисло". Первый из них обращается к проверочному оператору "усмне0" и при успехе заменяет равенство на константу "ложь". Равенство $b = c$ также обрабатывается нормализатором "нормчисло". Наконец, вся дизъюнкция обрабатывается нормализатором "нормлог". Таким образом, если очевидно, что $a \neq 0$, то заменяющий терм второго приема будет отличаться от заменяющего терма первого лишь дополнительной стандартизацией. Уровень срабатывания обоих приемов равен 2.

3. Равенство с совпадающими частями.

   $\forall_a(a = a)$

   Указатель "эквивалентно" поясняет, что имеется в виду не вырожденное тождественное преобразование, а замена равенства на константу "истина". Уровень срабатывания равен 1.

4. Равенство с минусом в одной части и нулем в другой.

   $\forall_a(a - \text{число} \to -a = 0 \leftrightarrow a = 0)$

5. Отбрасывание минусов в обеих частях равенства.

   $\forall_{ab}(a - \text{число} \mathbin{\&} b - \text{число} \to -a = -b \leftrightarrow a = b)$

6. Равенство степени нулю.

   $\forall_{ab}(a^b = 0 \leftrightarrow a = 0)$

7. Равенство дроби нулю.

   $\forall_{ab}(\neg(b = 0) \to a = 0 \leftrightarrow a/b = 0)$

   Прием имеет заголовок "замена(первыйтерм нормчисло)".

8. Равенство произведения нулю.

   $\forall_{ab}(ab = 0 \leftrightarrow a = 0 \ \vee \ b = 0)$

9. Равенство модуля нулю.

   $\forall_a(a - \text{число} \to |a| = 0 \leftrightarrow a = 0)$

10. Перестановка неизвестной части равенства влево.

    $\forall_{ab}(a = b \leftrightarrow b = a)$

    Выражение a не содержит неизвестных, а выражение b - содержит. Указатель "коммутативно(фикс(0 1))" блокирует перестановку частей равенства при его идентификации.

Нормализатор выявления повторяющихся вхождений подвыражений с неизвестными "повторчисло"

Этот нормализатор упоминался выше в связи с реализованными на ЛОСе приемами, обеспечивающими переход к новым неизвестным. Он преобразует заданное утверждение (уравнение либо неравенство) таким образом, чтобы выявилось возможно большее количество повторяющихся вхождений одинаковых подвыражений с неизвестными. Если преобразуется уравнение, причем задача имеет также другие уравнения, то нормализатору передается комментарий (внешвхождение A), где A - список остальных уравнений. Этот список позволяет учитывать возможность повторных вхождений в различные уравнения. Если в процессе преобразований текущего уравнения становится необходимым вернуться к преобразованиям какого-либо другого уравнения списка A, то оно регистрируется в комментарии (новыенеизвестные ...). Последний был создан при обращении к нормализатору. Нормализатор не является корневым, т.е. приводимые ниже приемы применяются к произвольному выделяемому в преобразуемом терме вхождению. Заметим, что для компенсации неудачных переходов предусмотрены преобразования перегруппировки. Это позволяет обходиться без трудоемкого перебора множества альтернативных группировок.

Перейдем к рассмотрению приемов нормализатора.

1. Простейшая стандартизация.

   В разделе собраны простейшие стандартизирующие приемы, сопровождающие основные преобразования нормализатора. Их список весьма ограничен, так как более сильная стандартизация может нарушить идентичность специально выделенных подвыражений.

   (a) Знаменатель единица.
   $\forall_a(a/1 = a)$

   (b) Множитель единица.
   $\forall_a(a \cdot 1 = 1)$

   (c) Лексикографическое упорядочение слагаемых.

   (d) Лексикографическое упорядочение сомножителей.

   (e) Приведение подобных членов с известными коэффициентами.
   $\forall_{abc}(ac + bc = (a + b)c)$

   Переменная c идентифицируется с операндом операции "умножение". Это обеспечивается указателями "операнд(x3 фикс(0 1 1))" и "наборoperandов(фикс(0 1 2))". Проверяется, что c содержит неизвестные и что a, b известны. Иногда оказывается, что в произведении нужно выделить подпроизведение, имеющее другое вхождение в преобразуемые условия задачи. Такое подпроизведение "заключается в скобки", иными словами, искусственно создаются вложенные операции "умножение". Чтобы прием не нарушал этой группировки, введен указатель "спускоперандов(x3)". Он блокирует устранение вложенных умножений для заменяющего терма, сохраняя в нем заголовок "умножение" выражения c, если таковой имелся изначально. Выражение $a + b$ обрабатывается нормализатором "нормплюс". Нормализатор "нормумножение" может нарушить выделение идентичных подтермов и не применяется.

(f) Вынесение знака "минус" из сомножителя.

$\forall ab((-a)b = -ab)$

(g) Устранение повторной степени.

$\forall_{abc}(c - \mathrm{rational}\ \&\ \neg(\text{знаменатель}(c) - \mathrm{even}) \rightarrow (a^b)^c = a^{bc})$

Чтобы не нарушить ранее достигнутого выделения повторяющихся вхождений, введен фильтр "не(контекст(внешвхождение(x4)вид(x4 степень(x1 x5))или(равно(x2 x5)не(известно(x5)))))". Оператор "внешвхждение(x4)" перечисляет все вхождения подтермов, расположенные вне текущего вхождения - как в обрабатываемый нормализатором терм, так и в термы комментария "внешвхождение". Если среди них обнаруживается либо вхождение выражения a^b, либо вхождение выражения вида a^e, где e не известно, то прием блокируется. Нормализатором общей стандартизации обрабатывается лишь подтерм bc заменяющего терма. Нормализатор "нормстепень" не используется, так как может нарушить выделение идентичных подтермов.

(h) Показатель степени единица.

$\forall_a(a^1 = a)$

(i) Минус в показателе степени.

$\forall_{ab}(a^{-b} = 1/a^b)$

Применение приема сильно ограничено. Во-первых, требуется отсутствие "внешних" подвыражений вида a^{bc}, в том числе с минусом в показателе степени. Во-вторых, выражение a должно содержать тригонометрическую операцию с неизвестными. Последнее требование объясняется тем, что если новые неизвестные вводятся для выражений с тригонометрическими операциями, то можно пренебречь небольшими рассогласованиями в их надвыражениях, возникающими при применении стандартизирующих тождеств с более "простыми" степенными операциями.

(j) Умножение на дробь.

$$\forall_{abc}(\frac{a}{b}c = \frac{ac}{b})$$

Не допускается наличие "внешних" подвыражений, равных a/b либо полученных из c отбрасыванием части известных множителей. При этом требуется наличие как в a/b, так и в c тригонометрической операции с неизвестными.

(k) Деление дроби.

$\forall_{abc}((a/b)/c = a/(bc))$

Не допускается наличие "внешних" подвыражений, равных a/b либо c. Каждое из последних должно содержать тригонометрическую операцию с неизвестными.

(l) Отбрасывание ненулевых множителей левой части уравнения с нулевой правой частью.

$\forall_{ab}(\neg(a = 0) \rightarrow ab = 0 \leftrightarrow b = 0)$

Попытка применения приема выполняется, если число неизвестных текущей задачи более одной, причем оба множителя a, b имеют неизвестные.

Имеется достаточно сильный ограничитель трудоемкости обработки антецедента проверочным оператором.

(m) Перемножение степеней с одинаковыми основаниями.

$\forall_{abc}(b$—rational & c—rational & $\neg$(знаменатель(b)—even & $\neg$(знаменатель(c)—even) $\rightarrow a^b a^c = a^{b+c})$

Выражение a имеет неизвестные. Уровень срабатывания равен 2.

(n) Устранение избыточной повторной степени.

$\forall_{abc}((a^b)^c = a^{bc})$

Выражения b, c не содержат неизвестных. Либо a имеет заголовок "степень", либо отсутствует внешнее вхождение выражения a^b. Последнее ограничение предотвращает разрушение ранее достигнутого специального выделения подвыражения a^b. Уровень срабатывания равен 6.

(o) Вынесение наружу известных множителей основания степени

$\forall_{abc}(0 \leq a \ \& \ 0 \leq b \rightarrow (ab)^c = a^c b^c)$

$\forall_{abc}(c$ – rational & $\neg$(знаменатель(c) – even) $\rightarrow (ab)^c = a^c b^c)$

Выражение a идентифицируется с произведением всех известных множителей. Проверяется невырожденность выражений a, b. Уровень срабатывания равен 5.

2. Группировка в одной части уравнения всех неизвестных слагаемых.

Такая группировка может выявить повторное вхождение суммы, ранее распределенной между разными частями равенства.

$\forall_{ab}(a$ – число & b – число $\rightarrow a = b \leftrightarrow a - b = 0)$

Проверяется, что выражения a, b содержат неизвестные и что равенство - корневое.

3. Группировка слагаемых для выделения повторного вхождения.

Если несколько неизвестных слагаемых одной суммы составляют все неизвестные слагаемые другой суммы, то предпринимается их группировка.

$\forall_{abcd}(c = a + d \rightarrow a + b = a + b)$

Указатель "контекст(другоевхождение(x3))" идентифицирует некоторую "внешнюю" сумму c либо собственную подсумму текущей суммы. Первый антецедент выделен указателем "идентификатор". Он нужен для определения общей части a слагаемых текущей суммы и суммы c. Указатель "выписка(x1 не(известно(x1)))" уточняет, что общая часть берется только для неизвестных слагаемых. Фильтры приема проверяют, что сумма d остальных слагаемых выражения c не содержит неизвестных; что число слагаемых выражения a более одного; что текущая сумма не совпадает с выражением c и что текущая сумма имеет не менее трех слагаемых. Указатель "единица(0 x4)" разрешает вырождение фрагмента d. Однако, для фрагмента b такого указателя нет, и он должнен быть невырожденным. Указатель "спускоперандов(x1)" "защищает" группировку подсуммы a, блокируя устранение в заменяющем терме вложенных сумм.

4. Группировка неизвестных квадратов.

Раздел содержит несколько приемов, усматривающих возможность выявления повторного вхождения суммы квадратов двух неизвестных выражений.

$\forall_{abcde}(a(b^2+e)+a(c^2+d)=a(b^2+c^2+e+d))$

Для учета повторного вхождения введен фильтр "контекст(внешвхождение(x6) или(контекст(вид(x6 плюс(степень(x2 2)степень(x3 2)x8))единица(0 x8)) контекст(вид(x6 плюс(минус(степень(x2 2))минус(степень(x3 2))x8))единица(0 x8)))буфер(новыенеизвестные базавхождения(x6)))".

Переменная x6 идентифицируется с некоторой внешней суммой, имеющей своими слагаемыми b^2, c^2, быть может, со знаками минус. Указатель "буфер(новыенеизвестные базавхождения(x6))", добавленный внутри фильтра, регистрирует в комментарии (новыенеизвестные ...) то уравнение, к которому относится сумма x6. Это уравнение будет рассмотрено повторно.

Выражения b, c содержат неизвестные, причем общее число неизвестных задачи не менее двух. Уровень срабатывания приема равен 3.

$\forall_{abcd}(-(a^2+b)-(c^2+d)=-(a^2+c^2+b+d))$

Прием аналогичен предыдущему, но уровень его срабатывания равен 4.

Далее идут два приема, теоремы которых идентичны только что приведенным. Отличие состоит в учете повторного вхождения: внешняя сумма имеет своими слагаемыми не квадраты выражений, а сами эти выражения. Так подготавливается выделение квадрата суммы данных выражений.

5. Группировка неизвестных кубов.

   Сумма либо разность кубов выделяются с помощью следующих приемов:

   $\forall_{abcde}(a(b^3+e)+a(c^3+d)=a(b^3+c^3+e+d))$

   $\forall_{abcde}(a(b^3+e)+a(-c^3+d)=a(b^3-c^3+e+d)$

   Как и в случае группировки квадратов, требуется наличие внешней суммы, имеющей своими слагаемыми степени выражений b, c с одинаковыми показателями. Допускается вырожденный случай равенства показателя единице. Уровень срабатывания приемов равен 3.

6. Сумма и разность квадратов.

   Раздел содержит несколько приемов, выражающих сумму либо разность квадратов двух величин через сумму, разность и произведение этих величин. Во всех случаях предполагается, что обе величины не известны, а число неизвестных задачи не менее двух.

   (a) Выражение суммы квадратов через сумму и произведение.

      Имеются два приема с одной и той же теоремой:

      $\forall_{bc}(b^2+c^2=(b+c)^2-2(bc))$

      В первом приеме требуется наличие внешнего вхождения произведения, имеющего множители b, c, а также внешнего вхождения суммы, имеющей слагаемые kb, kc. Коэффициент k может обращаться в единицу. Эти внешние вхождения регистрируются в буфере "новыенеизвестные". Отбрасывается случай, когда рассматриваемая сумма квадратов сама возводится в квадрат, причем имеются внешние вхождения произведения $A \cdot p \cdot (b^2+c^2)$

и суммы $B + kp + k(b^2 + c^2)$; здесь p равно либо b, либо c. В этом случае сумма квадратов сохраняется, чтобы ее обозначить новой неизвестной.

Фильтр "лексикопредшествует(x2 x3)" введен для отсечения симметричных ситуаций. Прием срабатывает на уровне 3. Обращаем внимание на то, что произведение bc, обрабатываемое нормализатором "нормумножение", сгруппировано внутри внешнего произведения. Сумма $b+c$ обрабатывается нормализатором "нормплюс", и других обращений к нормализаторам прием не имеет.

Второй прием срабатывает на уровне 5. Для его срабатывания необходимо либо наличие внешнего вхождения произведения, имеющего множители b, c, либо наличие внешнего вхождения суммы, имеющей слагаемые kb, kc. Напомним, что первый прием требовал и того, и другого одновременно. Далее, второй прием проверяет отсутствие внешних вхождений кубов выражений b, c. Его применение блокируется, если каждое выражение b, c имеет вид степени, причем существует внешнее вхождение равенства суммы b и c известному выражению. Комментарий (суммавсех 2), означающий, что ранее сумма квадратов неизвестных величин выражалась через сумму и разность этих величин, также блокирует применение данного приема.

(b) Выражение суммы квадратов через разность и произведение.
$\forall_{ab}(a^2 + b^2 = (a - b)^2 + 2(ab))$

Требуется наличие внешнего вхождения произведения, имеющего множители a, b, а также внешнего вхождения суммы, имеющей слагаемые $ka, -kb$. Коэффициент k может обращаться в единицу. Указанные вхождения регистрируются в буфере "новыенеизвестные". Уровень срабатывания приема равен 3.

(c) Выражение суммы квадратов через сумму и разность.

$$\forall_{ab}(a^2 + b^2 = \frac{(a - b)^2 + (a + b)^2}{2})$$

Требуется наличие внешнего вхождения суммы со слагаемыми $ma, -mb$, а также внешнего вхождения суммы со слагаемыми na, nb. Каждая из сумм должна иметь не менее двух неизвестных. Условия, содержащие данные суммы, регистрируются в буфере "новыенеизвестные". Прием срабатывает на уровне 4. Он вводит комментарий (суммавсех 2), блокирующий альтернативные преобразования суммы квадратов.

(d) Выражение разности квадратов через сумму и разность.
$a^2 - b^2 = (a - b)(a + b)$
$a^2 - b^2 = -(b - a)(a + b)$

В первом случае имеется внешнее вхождение суммы со слагаемыми $a, -b$, во втором - суммы со слагаемыми $b, -a$. В обоих случаях, кроме того, имеется внешнее вхождение суммы со слагаемыми a, b. Уровень срабатывания приема равен 3.

7. Сумма и разность кубов.

   (a) Выражение суммы кубов через сумму и произведение.

$\forall_{bc}(b^3 + c^3 = (b + c)((b + c)^2 - 3(bc)))$

Созданы две версии приема. Для одной необходимо наличие внешнего вхождения суммы со слагаемыми mb^2, mc^2, для другой - суммы со слагаемыми mb, mc. Кроме того, требуется внешнее вхождение произведения с сомножителями b, c. Уровень срабатывания приемов равен 4.

(b) Выражение разности кубов через разность и произведение.
$\forall_{ab}(a^3 - b^3 = (a - b)((a - b)^2 + 3(ab)))$
$\forall_{ab}(a^3 - b^3 = -(b - a)((b - a)^2 + 3(ab)))$

В первом случае требуется внешнее вхождение суммы со слагаемыми ma, $-mb$, во втором - суммы со слагаемыми $mb, -ma$. В обоих случаях необходимо также наличие внешнего произведения с сомножителями a, b. Уровень срабатывания приемов равен 4. Имеется еще один прием, основанный на первой из указанных выше теорем. Для его применения необходимо наличие внешнего вхождения суммы со слагаемымми ma^2, mb^2, а также внешнего вхождения произведения с сомножителями a, b. Уровень срабатывания равен 5.

8. Выражение суммы четвертых степеней через сумму и произведение.

$\forall_{ab}(a^4 + b^4 = ((a + b)^2 - 2(ab))^2 - 2(ab)^2)$

Требуется наличие внешней суммы со слагаемыми ma, mb и внешнего произведения с сомножителями a, b. Уровень срабатывания равен 4.

9. Попытка разложения на множители суммы, содержащей неизвестные.

Предпринимается попытка применить упрощенную процедуру разложения на множители к группе неизвестных слагаемых некоторой суммы. Если при этом обнаруживается повторное вхождение выражения с неизвестными, то преобразование реализуется. Теорема приема имеет вид:

$\forall_{abc}(c = a \rightarrow a + b = b + c)$

Указатель "перечень(x1 не(известно(x1)))" определяет идентификацию a с суммой всех неизвестных слагаемых выражения $a + b$. Проверяется, что число таких слагаемых более одного. Антецедент, выделенный указателем "идентификатор", обращается к пакетному нормализатору "факторизация". Этот нормализатор содержит лишь самые простые приемы разложения на множители, например, вынесение за скобку общего множителя всех слагаемых. Переменной c присваивается результат обращения. Проверяется, что c имеет либо несколько множителей с неизвестными, либо один степенной множитель с неизвестными. Проверяется наличие либо внешнего произведения, имеющего общий неизвестный множитель с c, либо такой внешней суммы, что применение к подсумме ее неизвестных слагаемых нормализатора "факторизация" дает результат, имеющий общий неизвестный множитель с c. Перед всеми этими действиями проверяется, что выражение a не является линейным относительно неизвестных текущей задачи. Проверяется, что преобразуемая сумма не является показателем степени. Наконец, проверяется отсутствие внешней суммы, подмножество неизвестных слагаемых которой равно a. В случае текущей задачи на исследование прием блокируется. Уровень срабатывания равен 5.

10. Выделение произведения двух неизвестных сумм.

Если сумма имеет три неизвестных слагаемых, возникающих при перемножении двух неизвестных двучленов, то в ней выделяется "полное" произведение: $\forall_{abcde}(ab+bc+ad+e=(b+d)(a+c)+e-cd)$

Предполагается, что выражения a,b содержат неизвестные, а c,d - не содержат. Фильтры "контекст(другоевхождение(x6)вид(x6 плюс(x2 x4)))", "контекст(другоевхождение(x6)вид(x6 плюс(x1 x3)))" проверяют существование внешних вхождений выражений $a{+}c$, $b{+}d$ либо вхождений их, расположенных строго внутри рассматриваемой суммы.Уровень срабатывания равен 5.

11. Перегруппировка неизвестных слагаемых.

Если в некоторой сумме были выделены две неизвестные подсуммы, хотя бы одна из которых не имеет прочих вхождений, причем перегруппировка слагаемых этих подсумм дает две уже встречавшиеся другие подсуммы, то такая перегруппировка выполняется. Теорема приема имеет вид:

$\forall_{abcdefgh}(c=a+d\ \&\ f=b+e\ \&\ g=d+e\rightarrow(a+b)+g+h=c+f+h)$

Указатель "контекст(другоевхождение(x3)другоевхождение(x6))" определяет идентификацию выражений c,f с некоторыми внешними суммами либо с собственными подсуммами текущей суммы $(a{+}b){+}g{+}h$. После этого первые два антецедента проверяют, что некоторое слагаемое текущей суммы разбивается на два фрагмента a,b сумм c,f. Оставшиеся части d,e последних сумм объединяются и обрабатываются нормализатором "нормплюс". Третий антецедент присваивает результат данного преобразования переменной g, идентифицируемой с отдельным слагаемым текущей суммы. Проверяется, что суммы c,f различны и не имеют известных слагаемых. Проверяется также, что их вхождения не расположены внутри слагаемого $(a+b)$. Наконец, проверяется отсутствие другого вхождения подвыражения g. Указатели "спускоперандов(x3)", "спускоперандов(x6)" блокируют устранение вложенных подсумм c,f заменяющей суммы. Уровень срабатывания равен 6.

12. Вынесение наружу общего знака "минус" у группы неизвестных слагаемых.

$\forall_{ab}(b=-a\rightarrow a=-b)$

Выражение a представляет собой сумму, имеющую более одного неизвестного слагаемого, причем все такие слагаемые берутся со знаком "минус". Первый антецедент находит результат b изменения знаков у всех слагаемых a. Уровень срабатывания равен 2.

В разделе имеется еще один прием, усматривающий, что при изменении знаков слагаемых можно получить ранее встречавшееся выражение. Теорема его такова:

$\forall_{abcde}(-a=b+d\ \&\ c=b+e\rightarrow a=-b-d)$

Указатель "контекст(другоевхождение(x3))" определяет идентификацию переменной c с внешней суммой либо с собственной подсуммой текущей суммы. Первый антецедент определяет результат замены у суммы a всех знаков на противоположные, после чего он, совместно со вторым антецедентом, выявляет общую часть b полученной суммы и суммы c. Указатель "выписка(x2 не(известно(x2)))" определяет отнесение к b лишь тех общих слагаемых, которые содержат неизвестные. Проверяется, что как a, так и c имеют более одного неизвестного слагаемого, причем хотя бы одно из них - со знаком "минус".

Проверяется, что a не имело ранее выделенного неизвестного слагаемого, имеющего вид суммы с вынесенным наружу знаком "минус". Наконец, проверяется, что либо выражения d, e известны, либо одна из сумм a, c расположена внутри другой. Уровень срабатывания равен 3.

13. Группировка сомножителей для получения повторного вхождения.

    Прием аналогичен приему группировки слагаемых:

    $\forall_{ab}(c = ad \to ab = ab)$

    Указатель "контекст(другоевхождение(x3))" идентифицирует c; далее a определяется как совокупность общих неизвестных множителей выражения c и текущего произведения. Проверяется, что d не содержит неизвестных и что a имеет не менее двух сомножителей. Уровень срабатывания равен 3.

14. Вынесение за скобку общего известного множителя нескольких неизвестных слагаемых.

    Вынесение за скобку общего известного множителя группы неизвестных слагаемых является предварительной стандартизацией, увеличивающей вероятность обнаружения повторных вхождений. Первый из приемов, обеспечивающих такое вынесение, относится к сумме, имеющей ровно два неизвестных слагаемых:

    $\forall_{abc}(ab + ac = a(b + c))$

    Здесь a идентифицируется с произведением всех общих известных множителей двух неизвестных слагаемых. Уровень срабатывания равен 3. Второй прием основан на той же самой теореме. Его указатели "набор(второйтерм)", "перечень(x1 известно(x1))" определяют идентификацию общего известного множителя a всех слагаемых текущей суммы. Блокируется случай, когда эта сумма - показатель степени. Уровень срабатывания равен 1.

15. Вынесение за скобку общего неизвестного множителя двух неизвестных слагаемых.

    Теорема приема та же, что в предыдущем разделе:

    $\forall_{abc}(ab + ac = a(b + c))$

    Выражения a, b, c содержат неизвестные, причем имеется внешнее вхождение суммы одинаковых степеней выражений b, c либо выражений $-b, -c$. Уровень срабатывания равен 2.

    Имеется еще один аналогичный прием:

    $\forall_{abc}(ab^2 + ac^2 = a(b^2 + c^2))$

    Как и выше, выражения a, b, c содержат неизвестные, причем должно найтись внешнее вхождение суммы одинаковых степеней выражений b, c. Уровень срабатывания равен 3.

16. Усмотрение полного квадрата в произведении нескольких сомножителей.

    $\forall_{abcd}(b = d^2 \to ab + c = ad^2 + c)$

    Выражение b идентифицируется с произведением всех неизвестных множителей некоторого слагаемого, причем таких множителей оказывается более одного. Антецедент усматривает, что b является квадратом некоторого выражения

d (компилятор использует процедуру "выделениестепени"). Проверяется существование внешнего вхождения произведения, группа всех неизвестных сомножителей которого равна d. Уровень срабатывания равен 5.

17. Устранение вложенных умножений при наличии известных сомножителей.

    $\forall_{abc}((ab)c = abc)$

    Если подпроизведение (ab) имеет известный множитель a и неизвестный множитель b, то оно сливается с внешним произведением. Уровень срабатывания равен 2.

18. Усмотрение в произведении четырех линейных множителей произведения двух квадратных трехчленов, отличающихся только свободными членами.

    Прием используется в некоторых школьных задачах, которые без него приводят к уравнениям четвертой степени.

    $\forall_{abcdpqrsnx}(n = dp + cq \;\&\; arn - cp(br + as) = 0 \rightarrow (ax + b)(cx + d)(px + q)(rx + s) = (ar/cp)(cpx^2 + nx + (cpbs/ar))(cpx^2 + nx + dq))$

    Выражения a, b, c, d, p, q, r, s не содержат неизвестных, выражение x - содержит. Антецеденты проверяют условия, необходимые для совпадения двух старших коэффициентов выделенных квадратных трехчленов. После данного преобразования становится возможным переход к новой неизвестной $y = cpx^2 + nx$. Уровень срабатывания равен 5.

19. Перестановка неизвестных множителей числителя и знаменателя.

    Прием усматривает две неизвестные взаимно обратные дроби и "переворачивает" одну из них:

    $$\forall_{abcd}\left(\frac{ac}{bd} = \frac{a}{b\frac{d}{c}}\right)$$

    Выражения c, d идентифицируются с произведениями всех неизвестных множителей числителя и знаменателя. Проверяется, что эти произведения невырожденные и что существует внешнее вхождение дроби, числитель которой делится на d, а знаменатель - на c. Уровень срабатывания равен 1.

20. Выделение неизвестных множителей числителя и знаменателя.

    Прием выносит наружу известные множители, выделяя отношение неизвестных выражений "в чистом виде":

    $$\forall_{abcd}\left(\frac{ac}{bd} = \frac{a\frac{c}{d}}{b}\right)$$

    Выражения c, d идентифицируются с произведениями всех неизвестных множителей числителя и знаменателя. Проверяется их невырожденность, отличие от единицы хотя бы одного из выражений a, b, а также существование внешнего вхождения дроби, числитель которой делится на c, а знаменатель на d. Уровень срабатывания равен 2.

Следующий прием ориентирован на выделение неизвестного подвыражения вида $1/c$:

$$\forall_{abc}\left(\frac{a}{bc}=\frac{a}{b}\cdot\frac{1}{c}\right)$$

Числитель a известен; хотя бы одно из выражений a, b отлично от 1; выражение c идентифицируется с произведением всех неизвестных множителей знаменателя. Требуется наличие внешнего вхождения дроби, у которой произведение всех неизвестных множителей знаменателя равно c, причем либо числитель известен, либо задача имеет более одной неизвестной. Требуется также, чтобы каждое внешнее вхождение выражения c представляло собой сомножитель основания степени сомножителя знаменателя некоторой дроби.

Наконец, в разделе имеется прием, подразбивающий неизвестную дробь в произведение таких двух неизвестных дробей, одна из которых уже встречалась во внешнем контексте:

$$\forall_{abcdefgh}\left(a=bc\ \&\ d=ef\rightarrow\frac{ga}{hd}=\frac{g}{h}\cdot\frac{b}{e}\cdot\frac{c}{f}\right)$$

Выражения a, d идентифицируются со всеми неизвестными множителями числителя и знаменателя. Указатель "контекст(внешвхождение(x9)вид(x9 дробь(умножение(x10 x3)умножение(x11 x6)))единица(1 x10 x11))" определяет выделение некоторой внешней дроби, причем c идентифицируется с общей частью ее числителя и выражения a, а f - с общей частью ее знаменателя и выражения d. Уровень срабатывания равен 4.

21. Степень дроби с числителем единица.

Если специальное выделение неизвестной дроби $1/a$, встречающейся в основании степени, оказывается ненужным, то выполняется возведение в степень числителя и знаменателя:

$$\forall_{ab}\left(b-\mathrm{rational}\ \&\ \neg(\text{знаменатель}(b)-\mathrm{even}\rightarrow\left(\frac{1}{a}\right)^{b}=\frac{1}{a^{b}}\right)$$

$$\forall_{ab}\left(0<a\rightarrow\left(\frac{1}{a}\right)^{b}=\frac{1}{a^{b}}\right)$$

Первый прием применяется, если показателем степени является рациональное число с нечетным знаменателем, второй - если усматривается положительность выражения a. В обоих случаях проверяется существование внешнего вхождения выражения a, не являющегося знаменателем дроби вида $1/a$. Уровень срабатывания равен 1.

22. Сложение дробей с известными знаменателями.

Если имеется сумма нескольких дробей с известными числителями, расположенная внутри степенного выражения, то выполняется сложение этих дробей:

$$\forall_{abcde}\left(\frac{a}{b}+c=\frac{d}{e}\rightarrow\frac{a}{b}+c=\frac{d}{e}\right)$$

Антецедент обращается для сложения дробей к нормализатору "видумножение". Проверяется наличие внешнего вхождения какой-либо суммы. Уровень срабатывания равен 4.

23. Усмотрение квадрата суммы неизвестного выражения и обратного ему выражения.

    Еще один часто используемый в школьных задачах трюк, позволяющий усмотреть скрытое вхождение неизвестного выражения $a + 1/a$:

    $$\forall_{ab}\left(ba^2 + \frac{b}{a^2} = b\left(\left(a + \frac{1}{a}\right)^2 - 2\right)\right)$$

    Выражение a содержит неизвестные, b - не содержит. Проверяется наличие внешнего вхождения суммы, имеющей слагаемые $a, 1/a$. Уровень срабатывания равен 3.

24. Переход от произведения неизвестных множителей к произведению их степеней.

    Уравнение с произведением двух неизвестных выражений a, b в одной из своих частей может быть возведено в степень, если каждое из этих выражений уже встречается в данной степени.

    $\forall_{abcd}(d{-}\mathrm{rational}\ \&\ \neg(\text{знаменатель}(d){-}\mathrm{even})\ \&\ g = a^d\ \&\ \neg(\text{числитель}(d){-}\mathrm{even}) \to ab = c \leftrightarrow a^d b^d = c^d)$

    $\forall_{abcd}(d - \mathrm{rational}\ \&\ \text{знаменатель}(d) - \mathrm{even}\ \&\ g = a^d\ \&\ 0 \le a\ \&\ 0 \le b\ \&\ a - \text{число}\ \&\ b - \text{число} \to ab = c \leftrightarrow a^d b^d = c^d)$

    Указатель "контекст(внешвхождение(x7))" определяет выбор внешнего вхождения некоторого выражения g. Третий антецедент обнаруживает, что сомножитель a является основанием степени выражения g. Далее проверяется наличие внешнего вхождения выражения b^d. Проверяется, что каждое внешнее вхождение выражений a, b представляет собой основание степени с показателем d. Уровень срабатывания приемов равен 2.

25. Группировка под общий известный показатель степени.

    Если перемножаются две степени неизвестных выражений a, b с общим известным показателем, причем имеется внешнее вхождение произведения P с сомножителями a, b, то предпринимается группировка степеней под общий показатель. Требуется, чтобы P не имело других неизвестных сомножителей.

    $\forall_{abc}(c - \mathrm{rational}\ \&\ \neg(\text{знаменатель}(c) - \mathrm{even}) \to a^c b^c = (ab)^c)$

    Уровень срабатывания равен 3.

26. Выделение повторной степени с неизвестным основанием.

    В разделе представлены приемы, преобразующие обычную степень в повторную. Возникающая при этом внутренняя степень уже встречается в рассматриваемых утверждениях. Уровень срабатывания приемов равен 5.

    $\forall_{abcn}(a - \text{число}\ \&\ b - \mathrm{rational}\ \&\ \text{знаменатель}(b) - \mathrm{even}\ \&\ n = (c/b)\ \&\ 0 \le a\ \&\ n - \text{целое}\ \&\ c - \mathrm{rational} \to (a^b)^n = a^c)$

В отличие от предыдущих теорем, здесь применяется замена справа налево. Указатель "контекст(внешвхождение(x4)вид(x4 степень(x1 x2))символ(x2 дробь))" обнаруживает внешнее вхождение степени a^b с дробным показателем b. Проверяется, что a содержит неизвестные и что преобразуемая степень a^c не является основанием внешней степени. Заметим, что допускается вырожденный случай равенства c единице, однако в этом случае проверяется ряд дополнительных условий. Четвертый антецедент определяет выражение n; проверяется отличие n от единицы.

$\forall_{abcn}(b - \text{rational} \ \& \ \neg(\text{знаменатель}(b) - \text{even}) \ \& \ n = (c/b) \ \& \ c - \text{rational} \ \& \ n -$ целое $\& \ a -$ число $\rightarrow (a^b)^n = a^c)$

Прием аналогичен предыдущему.

$\forall_a(a - \text{число} \ \& \ a \leq 0 \rightarrow a = -(\sqrt{-a})^2)$

Усматривается внешнее вхождение выражения $\sqrt{-a}$. Проверяется, что заменяемое выражение a содержит неизвестные и является сомножителем произведения, имеющего другие неизвестные множители.

$\forall_{ab}(0 \leq a - b \rightarrow b - a = -(\sqrt{a-b})^2)$

Аналогично предыдущему.

27. Представление степени произведения в виде произведения степеней.

    Приемы раздела преобразуют степень произведения неизвестных выражений в произведение степеней, если последние имеют независимые внешние вхождения. Проверяется, что данные внешние вхождения не допускают обратного преобразования, т.е. не являются сомножителями общего произведения.

    $\forall_{abc}(0 \leq a \ \& \ 0 \leq b \rightarrow (ab)^c = a^c b^c)$

    $\forall_{abc}(c - \text{rational} \ \& \ \neg(\text{знаменатель}(c) - \text{even}) \rightarrow (ab)^c = a^c b^c)$

    Уровень срабатывания равен 5.

28. Выделение повторной степени с неизвестным показателем.

    Если имеется более одного неизвестного показателя, с которым в рассматриваемых утверждениях встречается степень некоторого выражения a, то предпринимается попытка вынести наружу возможно большее число известных фрагментов таких показателей. Это повышает вероятность обнаружения повторных вхождений выражения с неизвестными.

    $\forall_{abcde}(0 < a \ \& \ b = cd \rightarrow a^{b+e} = (a^d)^c a^e)$

    Выражение b идентифицируется с суммой всех неизвестных слагаемых показателя; выражение d - с произведением всех неизвестных сомножителей выражения b. Проверяется, что либо e, либо c невырождено. Проверяется наличие внешнего вхождения степени выражения a с неизвестным показателем, отличным от $b + e$. Если число неизвестных задачи менее двух, то проверяется отсутствие степени с неизвестным показателем и основанием, отличным от a. Наконец, проверяется, что рассматриваемое выражение не является фрагментом уравнения вида $pa^{b+e} = qa^m$, где p, q известны. Здесь выделение повторных вхождений неизвестной степени не требуется, так как она устраняется путем логарифмирования уравнения.

    Теорема следующего приема относится к дробным показателям степени:

$$\forall_{abcdefgh}\left(0 < a \to a^{\frac{de}{fg}+h} = a^h\left(a^{\frac{e}{g}}\right)^{\frac{d}{f}}\right)$$

Прием аналогичен предыдущему, однако дополнительно требуется наличие внешнего вхождения степени вида $a^{pe/qg}$, где множители p, q известны. Уровень срабатывания обоих приемов равен 5.

Последний прием раздела основан на следующей теореме:

$$\forall_{abcde}\left(e = a^{\frac{b}{c}} \to a^{\frac{bd}{c}} = \left(a^{\frac{b}{c}}\right)^d\right)$$

Просматриваются внешние вхождения выражения e, представимые в виде $a^{b/c}$, где числитель b показателя степени делит числитель bd показателя текущей степени. Требуется, чтобы хотя бы одно из выражений b, c содержало неизвестные. Уровень срабатывания равен 1.

29. Раскрывание скобок в показателе степени.

    $\forall_{abcde}(0 < a \;\&\; e = b(c+d) \to a^{b(c+d)} = a^e)$

    Проверяется, что оба сомножителя b, $c+d$ содержат неизвестные и что имеется внешнее вхождение степени выражения a с неизвестным показателем, отличным от данного. Второй антецедент обращается к нормализатору "стандплюс", выполняющему раскрывание скобок. Уровень срабатывания равен 5.

30. Сумма в показателе степени.

    Если показатель степени имеет несколько неизвестных слагаемых, то каждое из них выносится в отдельную степень:

    $\forall_{abc}(0 < a \to a^{b+c} = a^b a^c)$

    Здесь b - произвольное неизвестное слагаемое показателя; c - сумма остальных слагаемых. Проверяется наличие внешнего вхождения степени с основанием a, показатель которой не имеет подтерма $b + c$. Проверяется также, что рассматриваемое вхождение не является фрагментом уравнения вида $pa^{b+c} = q$, где p, q известны. Уровень срабатывания равен 4.

31. Перестановка числителя и знаменателя в основании степени с неизвестным показателем.

    $$\forall_{abc}\left(\left(\frac{a}{b}\right)^c = \left(\frac{b}{a}\right)^{-c}\right)$$

    Выражение c содержит неизвестные. Проверяется наличие внешнего вхождения степени дроби b/a, имеющей неизвестный показатель. Уровень срабатывания равен 3.

32. Переход к степени с дробным основанием.

    Приемы преобразуют отношение степеней с равными основаниями в степень дроби:

$$\forall_{abcde}\left(0 < a \;\&\; 0 < c \to \frac{da^b}{ec^b} = \frac{d\left(\frac{a}{c}\right)^b}{e}\right)$$

$$\forall_{abcde}\left(b - \mathrm{rational} \;\&\; \neg(\text{знаменатель}(b) - \mathrm{even}) \to \frac{da^b}{ec^b} = \frac{d\left(\frac{a}{c}\right)^b}{e}\right)$$

Первый прием относится к случаю неизвестного показателя степени b. Он проверяет наличие внешнего вхождения степени дроби a/c либо дроби c/a. Второй прием относится к случаю неизвестных выражений a, c. Здесь проверяется наличие внешнего вхождения дроби вида pa^b/qc^b. Уровень срабатывания приемов равен 3.

33. Две степени с неизвестными дробными показателями, дающими в сумме выражение без неизвестных.

    Если две степени с общим основанием имеют дробные неизвестные слагаемые в показателях, то при сложении показателей возможно сокращение неизвестных. В этом случае одна из степеней легко сводится к другой без использования новых степенных выражений с неизвестным показателем:

$$\forall_{abcdefgh}\left(\frac{b}{c} + \frac{e}{f} + d + g = h \to a^{\frac{b}{c}+d} = \frac{a^h}{a^{\frac{e}{f}+g}}\right)$$

    Сначала идентифицируется внешнее вхождение степенного выражения $a^{e/f+g}$. Затем антецедент с помощью нормализатора "видумножение" складывает показатели степени. Проверяется, что результат h не содержит неизвестных. Перед выполнением этих действий проверяется, что знаменатель c содержит неизвестные и имеет заголовок "плюс". Прием срабатывает на уровне 4.

34. Изменение основания степени при наличии логарифма в показателе.

    Прием позволяет преобразовать логарифмическое выражение к виду, уже имеющемуся в контексте:

$$\forall_{abcxy}\left(x^{\frac{a\log_c y}{b}} = y^{\frac{a\log_c x}{b}}\right)$$

    Выражения x, y содержат неизвестные. Имеется внешнее вхождение степенного выражения, показатель которого содержит $\log_c x$. Уровень срабатывания равен 5.

35. Выражение синуса двойного угла через сумму синуса и косинуса.

    $\forall_{ab}(a = b/2 \to \sin b = (\sin a + \cos a)^2 - 1)$

    Имеется внешнее вхождение суммы $\sin a + \cos a$. Выражение b содержит неизвестные. Либо число неизвестных более одной, либо преобразуемое выражение расположено под какой-либо "не алгебраической" операцией (экспонента, логарифм, синус, и т.д.). Антецедент упрощает выражение $b/2$ и затем сравнивает его с a. Уровень срабатывания равен 3.

36. Выражение синуса двойного угла через разность синуса и косинуса.

    $\forall_a(a = b/2 \to \sin b = 1 - (\sin a - \cos a)^2)$

    $\forall_a(a = b/2 \to \sin b = 1 - (\cos a - \sin a)^2)$

    Имеется внешнее вхождение соответствующей разности. Уровень срабатывания равен 3.

37. Выражение четной степени синуса (косинуса) через косинус (синус).

    $\forall_{ax}\left((\sin x)^a = (1 - (\cos x)^2)^{\frac{a}{2}}\right)$

    $\forall_{ax}\left((\cos x)^a = (1 - (\sin x)^2)^{\frac{a}{2}}\right)$

    a идентифицируется с четной константой. В первом случае проверяется, что каждое вхождение синуса x расположено под четной степенью, причем имеется внешнее вхождение косинуса x, расположенное под нечетной степенью. Второй случай рассматривается симметричным образом. Уровень срабатывания равен 3.

38. Выражение косинуса двойного угла через синус.

    $\forall_{ab}(a = b/2 \to \cos b = 1 - 2(\sin a)^2)$

    Имеется внешнее вхождение синуса a. Отсутствуют вхождения тангенса либо котангенса, а также синуса либо косинуса от другого ($\neq a, b$) аргумента, пересекающиеся по неизвестным с b. Уровень срабатывания равен 3.

39. Выражение косинуса двойного угла через косинус.

    $\forall_{ab}(a = b/2 \to \cos b = 2(\cos a)^2 - 1)$

    Аналогично предыдущему.

40. Применение формулы синуса (косинуса) суммы.

    $\forall_{ab}(\sin(a + b) = \sin a \cos b + \cos a \sin b)$

    Преобразуемое вхождение расположено в показателе степени и содержит неизвестные. Имеется внешнее вхождение тригонометрической операции, аргумент которой не является суммой и пересекается по своим неизвестным с a. Уровень срабатывания равен 3. Аналогичный прием имеется для косинуса.

41. Применение формулы тангенса (котангенса) суммы.

$$\forall_{ab}\left(\neg(\cos(a + b) = 0) \to \operatorname{tg}(a + b) = \frac{\operatorname{tg} a + \operatorname{tg} b}{1 - \operatorname{tg} a \operatorname{tg} b}\right)$$

$$\forall_{ab}\left(\neg(\sin(a + b) = 0) \to \operatorname{ctg}(a + b) = \frac{\operatorname{ctg} a \operatorname{ctg} b - 1}{\operatorname{ctg} a + \operatorname{ctg} b}\right)$$

В первом случае неизвестные $\operatorname{tg} a, \operatorname{tg} b$, а во втором - неизвестные $\operatorname{ctg} a, \operatorname{ctg} b$ должны иметь внешнее вхождение. Прием используется только при рассмотрении неравенств. Указатель "вывод(не(равно(плюс(1 минус(умножение(тангенс(x1) тангенс(x2))))0))эквивалентно(1))" пополняет сопровождение по о.д.з. утверждением об отличии знаменателя новой дроби от нуля. Уровень срабатывания равен 4.

42. Переход к тангенсу от линейной комбинации синуса и косинуса, деленной на косинус.

$$\forall_{abcdx}\left(\frac{d(a\sin x + b\cos x)}{c\cos x} = \frac{d(a\operatorname{tg} x + b)}{c}\right)$$

Преобразуемая дробь расположена в показателе нкоторой степени; имеется внешнее вхождение выражения $\operatorname{tg} x$. Уровень срабатывания равен 4.

43. Переход от произведения тригонометрических операций к сумме.

$\forall_{abcd}(c = \cos(a+b)\ \&\ d = \cos(a-b) \rightarrow \cos a\cos b = (c+d)/2)$

$\forall_{abcd}(c = \cos(a+b)\ \&\ d = \cos(a-b) \rightarrow \sin a\sin b = (d-c)/2)$

$\forall_{abcd}(c = \sin(a+b)\ \&\ d = \sin(a-b) \rightarrow \sin a\cos b = (c+d)/2)$

Выражения a, b содержат неизвестные, причем общее число неизвестных текущей задачи более одной. Каждое из выражений c, d имеет тригонометрическую операцию, аргумент которой совпадает с аргументом некоторого внешнего синуса либо косинуса. Существует такое внешнее вхождение неизвестного синуса либо косинуса, аргумент которого отличен от a, b. Уровень срабатывания равен 3.

44. Переход от суммы тригонометрических функций к произведению.

$$\forall_{abcde}\left(d = \cos\frac{a+b}{2}\ \&\ e = \sin\frac{a-b}{2} \rightarrow c\sin a - s\sin b = 2cde\right)$$

$$\forall_{abcde}\left(d = \cos\frac{a-b}{2}\ \&\ e = \sin\frac{a+b}{2} \rightarrow c\sin a + s\sin b = 2cde\right)$$

$$\forall_{abcde}\left(d = \sin\frac{a+b}{2}\ \&\ e = \sin\frac{a-b}{2} \rightarrow c\cos a - s\cos b = -2cde\right)$$

Каждое из выражений d, e имеет внешнее вхождение. Уровень срабатывания равен 3.

45. Переход от котангенса к тангенсу половинного аргумента.

$$\forall_{ab}\left(b = \frac{a}{2} \rightarrow \operatorname{ctg} a = \frac{1-(\operatorname{tg} b)^2}{2\operatorname{tg} b}\right)$$

Имеется внешнее вхождение выражения $\operatorname{tg} b$. Число неизвестных задачи более одной. Прием срабатывает на уровне 4.

46. Выражение котангенса через тангенс.

$$\forall_{a}\left(\operatorname{ctg} a = \frac{1}{\operatorname{tg} a}\right)$$

Имеется внешнее вхождение выражения $\operatorname{tg} a$. Уровень срабатывания равен 3.

47. Выражение четной степени тангенса через косинус.

$$\forall_{abc}\left(b = 2c \to (\operatorname{tg} a)^b = \left(\frac{1}{(\cos a)^2} - 1\right)^c\right)$$

Переменная b идентифицируется с четной целочисленной константой. Число неизвестных текущей задачи более одной. Имеется внешнее вхождение выражения $\cos a$. Прием срабатывает на уровне 4.

48. Переход к половинному углу при решении неравенств.

Следующие приемы относятся только к решению неравенств, на что указывает их общий фильтр "входит(уравнменьше комментарии)". Уровень срабатывания равен 3.

(a) Выражение синуса двойного угла через тангенс.

$$\forall_{ab}\left(b = 2a \to \sin b = \frac{2 \operatorname{tg} a}{1 + (\operatorname{tg} a)^2}\right)$$

Имеется внешнее вхождение выражения $\operatorname{tg} a$. Отсутствует вхождение синуса либо косинуса a, расположенное под четной степенью. Все неизвестные аргументы тригонометрических операций текущего неравенства кратны a с коэффициентами 1,2 либо 4.

(b) Выражение косинуса двойного угла через тангенс.

$$\forall_{ab}\left(b = 2a \to \cos b = \frac{1 - (\operatorname{tg} a)^2}{1 + (\operatorname{tg} a)^2}\right)$$

Аналогично предыдущему.

(c) Выражение четной степени синуса через тангенс.

$$\forall_{abc}\left(b = 2c \to (\sin a)^b = \left(\frac{(\operatorname{tg} a)^2}{1 + (\operatorname{tg} a)^2}\right)^c\right)$$

Аналогично предыдущему; b идентифицируется с целочисленной константой.

(d) Выражение четной степени косинуса через тангенс.

$$\forall_{abc}\left(b = 2c \to (\cos a)^b = \frac{1}{(1 + (\operatorname{tg} a)^2)^c}\right)$$

Аналогично предыдущему.

(e) Выражение синуса четырехкратного угла через синус и косинус двойного угла.

Прием подготавливает переход к тангенсу:

$\forall_{ab}(b = 4a \to \sin b = 2\sin(2a)\cos(2a))$

Имеется внешнее вхождение выражения $\operatorname{tg} a$. В остальном фильтры аналогичны предыдущим случаям.

(f) Выражение тангенса двойного угла через тангенс.

$$\forall_{ab}\left(b=2a\ \&\ \neg(\cos b=0)\rightarrow \operatorname{tg} b=\frac{2\operatorname{tg} a}{1-(\operatorname{tg} a)^2}\right)$$

Аналогично предыдущим приемам. Имеется указатель, регистрирующий в сопровождении по о.д.з. условие отличия от нуля нового знаменателя.

49. Выражение четной степени косинуса через тангенс.

$$\forall_{ax}\left((\cos x)^a=\frac{1}{(1+(\operatorname{tg} x)^2)^(a/2)}\right)$$

Переменная x является неизвестной. Имеется внешнее вхождение выражения $\operatorname{tg} x$, причем каждое внешнее вхождение переменной x расположено под тангенсом. a идентифицируется с четной целочисленной константой. Число неизвестных текущей задачи больше одной. Прием срабатывает на уровне 3.

Примеры применения оператора "повторчисло"

Приведем два простых примера, иллюстрирующих работу оператора "повторчисло". В первом из них решается система уравнений:

$$\begin{cases} x+y+z=1\\ xy+xz+yz=-4\\ x^3+y^3+z^3=1\end{cases}$$

Процедура перехода к новым неизвестным реализует следующий цикл обращений к оператору "повторчисло":

1. Анализируется уравнение $x+y+z=1$. Оно оставляется без изменений.

2. Анализируется уравнение $xy+xz+yz=-4$. Применяется группировка членов xy, xz: $yz+x(y+z)=-4$. Мотивировкой преобразования служит наличие суммы одинаковых степеней переменных y, z в первом и третьем уравнениях. Делаются пометки о необходимости повторного рассмотрения данных уравнений.

3. Анализируется уравнение $x^3+y^3+z^3=1$. Сумма кубов y^3, z^3 выражается через сумму и произведение: $x^3+(y+z)((y+z)^2-3(yz))=1$. Мотивировкой служит наличие внешних вхождений суммы $y+z$ и произведения yz. Делается пометка о необходимости повторного рассмотрения второго уравнения.

4. Снова анализируется первое уравнение $x+y+z=1$. Выделяется подсумма $y+z$, имеющаяся в остальных двух уравнениях: $x+(y+z)=1$.

5. Снова анализируется второе уравнение $yz+x(y+z)=-4$. Оно не изменяется.

По завершении цикла вводятся новые неизвестные u, v, w для выражений $y+z, yz, x$. Система приобретает следующий вид:

$$\begin{cases} w+u=1\\ wu+v=-4\\ u(-3v+u^2)+w^3=1\end{cases}$$

Дальнейший ход решения таков: с помощью первого уравнения исключается неизвестная u; полученные уравнения упрощаются. Одно из них распадается на два уравнения после попытки разложения на множители разности левой и правой части.

Вторым примером служит тригонометрическая система:

$$\begin{cases} (\sin x)^2 + (\cos y)^2 = \frac{11}{16} \\ \sin\frac{x+y}{2}\cos\frac{x-y}{2} = \frac{5}{8} \end{cases}$$

Здесь реализуется такой цикл обращений:

1. Анализируется первое уравнение. Оно оставляется без изменений.

2. Анализируется второе уравнение. В нем произведение тригонометрических операций заменяется на сумму: $\frac{\sin x+\sin y}{2} = \frac{5}{8}$. Мотивировкой служит наличие тригонометрический функций от x, y в первом уравнении. Делается пометка о необходимости повторного рассмотрения первого уравнения.

3. Снова рассматривается первое уравнение. В нем квадрат косинуса выражается через квадрат синуса, так как этот синус имеется во втором уравнении: $(\sin x)^2+(1-(\sin y)^2) = \frac{11}{16}$. Делается пометка о повторном рассмотрении второго уравнения.

4. Рассматривается второе уравнение. Оно не изменяется.

По завершении цикла вводятся новые неизвестные z, u для выражений $\sin y, \sin x$. Система становится совсем простой:

$$\begin{cases} (-z^2 + 1) + u^2 = \frac{11}{16} \\ \frac{u+z}{2} = \frac{5}{8} \end{cases}$$

Нормализатор "упрощповтор" нормализации выражений для вспомогательных неизвестных

После того, как были введены вспомогательные неизвестные и относительно них решена текущая задача, происходит возвращение к исходным неизвестным. При работе нормализатора "повторчисло" могли появиться группировки, нарушающие текущее сопровождение по о.д.з. для выражений, обозначенных вспомогательными неизвестными. Чтобы восстановить это сопровождение, указанные выражения обрабатываются нормализатором "упрощповтор". Он выполняет следующие действия: устраняет повторные степени; перемножает степени с одинаковыми основаниями; устраняет единичные показатели степени. Устраняются минусы перед разностью: $-(a-b) = b-a$. Если под четным радикалом возникает сумма, некоторое слагаемое которой имеет вид произведения на сумму, то выполняется раскрывание скобок. Таким образом восстанавливается тот вид выражения под радикалом, который используется в сопровождающем неравенстве, указывающем его неотрицательность.

Нормализатор равенств для параметров "стандчисло"

После того, как задача решена, может выполняться упрощение сопровождающих утверждений, содержащих только известные параметры. При этом используются специальные нормализаторы, одним из которых и является нормализатор "стандчисло". Ему передается для упрощения равенство, не содержащее неизвестных. В отличие от нормализаторов, обрабатывающих неравенства с параметрами, данный нормализатор несложен. Он имеет следующие приемы:

1. Явное разрешение равенства относительно параметра.

   $\forall_{abc}(\neg(a = 0) \to ab = c \leftrightarrow b = c/a)$

   Здесь c - константное выражение; a - все константные сомножители; b - невырожденное (т.е. неконстантное).

   $\forall_{abc}(a + b = c \leftrightarrow b = c - a)$

   Аналогично предыдущему.

   $\forall_{ab}(-b = a \leftrightarrow b = -a)$

   a - константное выражение, b - не константное.

   Все эти приемы срабатывают при наличии комментария "явное", указывающего на желательность явного разрешения относительно параметров. Уровень срабатывания равен 2.

2. Усмотрение ложности равенства, явно разрешенного относительно параметра.

   $\forall_{ab}(\neg(a = b) \to \neg(a = b))$

   Антецедент берется из списка посылок. a идентифицируется с переменной, b - с константой.

   $\forall_{abc}(a < b \ \& \ 0 \leq c - b \to \neg(a = c))$

   a идентифицируется с переменной; b, c - с константами. Первый антецедент берется из списка посылок, второй - обрабатывается проверочным оператором. Следующие три приема аналогичны:

   $\forall_{abc}(a \leq b \ \& \ 0 < c - b \to \neg(a = c))$

   $\forall_{abc}(b \leq a \ \& \ 0 < b - c \to \neg(a = c))$

   $\forall_{abc}(b < a \ \& \ 0 \leq b - c \to \neg(a = c))$

   Уровни срабатывания равны 1.

3. Усмотрение ненулевого значения константы.

   $\forall_{a}(\neg(a = 0) \to \neg(a = 0))$

   a идентифицируется с константным выражением. Антецедент обрабатывается проверочным оператором "усмне0". Уровень срабатывания равен 1.

4. Потенцирование простейших логарифмических равенств.

   $\forall_{abc}((0 < a) = \text{истина} \ \& \ (a - 1 = 0) = \text{ложь} \to \log_a b = c \leftrightarrow b = a^c)$

   Антецеденты выделены указателями "идентификатор". Левые их части обрабатываются нормализаторами сопровождающих утверждений (соответственно, "стандменьше" и "стандчисло"). Хотя бы одно из выражений a, b должно быть неконстантным. Необходимо также наличие комментария "явное". Заметим, что здесь мы сталкиваемся с таким явлением, как проверка условий $(0 < a, \neg(a - 1 = 0))$ не проверочными операторами, а нормализаторами, преобразующими их в логические константы "истина", "ложь". Причина состоит в том, что при завершающей обработке ответа сопровождение по о.д.з. может быть утеряно: сопровождающие утверждения упрощаются либо отбрасываются из-за своей логической избыточности. Обычные проверочные операторы здесь могут не сработать, в то время как нормализаторы сопровождающих утверждений легко справляются с задачей. Подробнее это явление будет рассмотрено в разделе, посвященном неравенствам.

5. Разбор случаев по дизъюнктивной посылке при усмотрении ложности равенства, явно разрешенного относительно переменной.

   $\forall_{abcdpq}((p \vee q)\ \&\ c = (a = b)\ \&\ c = \text{ложь}\ \&\ d = (a = b)\ \&\ d = \text{ложь} \rightarrow \neg(a = b))$

   a идентифицируется с переменной, b - с константой. Первый антецедент идентифицируется с дизъюнктивной посылкой, причем переменная a встречается как в p, так и в q. Остальные антецеденты выделены указателем "идентификатор". Второй и четвертый антецеденты обрабатывают равенство $a = b$ нормализатором "стандчисло" при разных списках посылок: в одном случае добавляется утверждение p, в другом - q. Вводимые при этом комментарии (подслучаи...) блокируют повторное применение данного приема. Если в обоих случаях равенство преобразуется к константе "ложь", то нормализатор заменяет его на данную константу. Уровень срабатывания равен 2.

Нормализатор "округлзнач"

Этот нормализатор имеет единственный прием, реализованный на ЛОСе. Он служит для приближенного вычисления значения константного выражения. При обращении к нормализатору вводится комментарий (число n), указывающий количество n знаков после запятой, до которого происходит округление. Пока данный нормализатор применяется только в задачах по химии.

10.4 Приемы символа "минус"

Выражение "минус(a)" обозначает результат изменения знака числа a. Перечислим приемы справочников, созданные для символа "минус". Справочник "тип" указывает, что операция "минус" принимает числовые значения; справочники "одз", "типданных" - что она определена для числовых аргументов; справочник "арность" - что операция одноместная. Справочник "область" указывает, что любой элемент из области определения операции входит в ее область значений; справочник "отрицание" - что операция является самообратной. Справочник "монотонно" указывает монотонное убывание значения операции. Справочник "вычисл", используемый компилятором ГЕНОЛОГа, определяет операторы ЛОСа для изменения знака числа, представленного в различных форматах. Справочники "минус", "неизвоценка", "заголовок" используются процедурами базы теорем.

Отбрасывание минуса перед нулем

Прием имеет бескванторную теорему $-0 = 0$ и срабатывает на уровне 0. Указатель "нормализатор" выделяет его как прием простейшей стандартизации; указатель "сопровождение" - разрешает применение даже в тех случаях, когда текущее утверждение используется для сопровождения по о.д.з.

Двойной минус

$\forall_a(--a=a)$

Аналогично предыдущему.

Отбрасывание минуса в равенстве нулю

$\forall_a(a=0\leftrightarrow -a=0)$

Прием имеет заголовок "первыйтерм". Уровень срабатывания равен 0. Введен указатель "сопровождение".

Минус перед разностью

$\forall_{ab}(-(b-a)=a-b)$

Заметим, что в логическом языке символ "вычитание" отсутствует. Поэтому разность понимается как сумма, имеющая слагаемые с заголовком "минус". Переменная a идентифицируется с каким-то одним таким слагаемым, b - с суммой остальных слагаемых. Чтобы $-b$ было преобразовано к стандартному виду суммы, выражения $-b$ и $a-b$ обрабатываются нормализаторами "нормминус" и "нормплюс". Первый из них необходим, если b имело заголовок "минус": он устранит двойной минус. Однако, нормализатор "нормминус" не заносит "минус" вглубь суммы, все слагаемые которой не имеют заголовка "минус". Последнее будет выполнено нормализатором "нормплюс", применяемым к $a-b$.

Прием выполняет общую стандартизацию и срабатывает на уровне 0. Для него введен указатель "нормализатор". Напомним, что этот указатель инициирует ускоряющий вычисления цикл попыток упрощения надвыражений сразу после преобразования текущего выражения. Ускорение достигается за счет экономии числа сканирований.

Равенство с минусом в одной из своих частей

Начнем с приема, перебрасывающего минус из неконстантной в константную часть равенства:

$\forall_{ab}(b-\text{число}\rightarrow -a=b\leftrightarrow a=-b)$

a идентифицируется с неконстантным выражением, b - с константным. Прием применяется на уровне 1 в условии задачи на описание (равенство не обязательно корневое). Для решения уравнений используется прием с такой же теоремой:

$\forall_{ab}(a-\text{число}\rightarrow a=-b\leftrightarrow b=-a)$

Здесь b содержит неизвестные, a - не содержит. Прием срабатывает на уровнях 0,1 в условии задачи на описание либо в задаче на исследование. В случае задачи на исследование, имеющей цель "известно", либо при рассмотрении дизъюнктивного утверждения берется уровень 0, иначе - 1. Фильтр "не(цель(редакция))" блокирует применение приема на этапе редактирования ответа. Фильтр "внешзнак(и или существует)" ограничивает срабатывания случаями, когда внешние по отношению к равенству символы суть только логические связки "и", "или" либо квантор существования: в других случаях сначала должны срабатывать приемы, упрощающие логический контекст. Наконец, применение приема блокируется, если равенство является частью уже полученного параметрического описания значений неизвестных задачи. Заметим, что проверка антецедента в данном преобразовании необходима: истинность исходного равенства гарантировала, что a имеет числовое значение. Вместе с тем, после преобразования эта информация, если она не была продублирована прочими утверждениями контекста, будет утеряна.

Следующий прием перебрасывает знак минус в противоположную часть равенства, чтобы получить явное выражение одной неизвестной через другие:

$\forall_{ab}(a - \text{число} \to -a = b \leftrightarrow a = -b)$

Прием применяется на уровне 0 в задачах на исследование. a - неизвестная, не встречающаяся в выражении b, причем b не является неизвестной. Прием блокируется в тех случаях, когда утверждение имеет числовые атомарные выражения, отличные от переменных и констант (например, "расстояние(...)", "угол(...)", и т.п.).

Наконец, приведем прием, выполняющий разрешение равенства относительно связанной переменной внешнего квантора:

$\forall_{ab}(a - \text{число} \to -a = b \leftrightarrow a = -b)$

Прием срабатывает на уровне 1. Переменная a встречается в связывающей приставке внешнего квантора, причем отсутствует промежуточное вхождение другого квантора, связывающего некоторые переменные выражения b.

Во всех перечисленных приемах перенесенный на новое место "минус" обрабатывается нормализатором "нормминус".

Усмотрение равенства суммы нулю, если равна нулю сумма с измененными знаками

$\forall_{ab}(a - b = 0 \to b - a = 0)$

Указатель "нормзнака(x2 минус плюс)" заставляет идентифицировать $-b$ в антецеденте как множество взятых с обратным знаком слагаемых суммы b. Таким образом обеспечивается усмотрение противоположных сумм с изменением знака у многих слагаемых. Указатель "эквивалентно" определяет прием, который будет заменять не выражение $b - a$ на 0, а равенство $b - a = 0$ на логическую константу "истина". Аналогичный прием для тождественной замены, требующий достаточно частых попыток применения, имеет более высокий уровень срабатывания. Фильтр "уровень(0 3)" обеспечивает попытку применения приема сначала на уровне 0, а затем (при измененном списке посылок, в котором уже может оказаться нужный антецедент) - на уровне 3.

Имеется еще одна версия данного приема - с той же теоремой, но без указателя "эквивалентно". Она срабатывает на уровнях 1 и 3.

$\forall_{ab}(\neg(a - b = 0) \to \neg(b - a = 0)$

Аналогично предыдущему приему, но без указателя "эквивалентно".

$\forall_{ab}((a - b = 0 \leftrightarrow b - a = 0) \leftrightarrow \text{истина})$

Аналогично предыдущему, но уровень срабатывания равен 1.

$\forall_{abP}(a - b = 0 \ \vee \ b - a = 0 \ \& \ P \leftrightarrow a - b = 0)$

Прием срабатывает на уровне 1. Указатель "дизъюнктоперанд" отменяет обобщенную идентификацию дизъюнкции.

Внесение минуса в сомножитель, имеющий вид разности

Этот прием обычно применяется на этапе завершающего редактирования, для более компактной формулировки результата.

$\forall_{abcde}(a - \text{rational} \ \& \ \neg(\text{числитель}(a) - \text{even}) \ \& \ \neg(\text{знаменатель}(a) - \text{even}) \rightarrow -(b - c)^a d/e = (c-b)^a d/e)$

Фильтр "или(и(тип(преобразовать) цель(учетрезультата)) и(тип(описать) цель(редакция)))" указывает на ситуацию редактирования ответа. Антецеденты обрабатываются проверочными операторами. Параметры a, d могут принимать вырожденные единичные значения. Уровень срабатывания приема равен 1. Следующие два приема аналогичны:

$\forall_{abcd}(a - \text{rational} \ \& \ \neg(\text{числитель}(a) - \text{even}) \ \& \ \neg(\text{знаменатель}(a) - \text{even}) \rightarrow -b(c-d)^a = b(d-c)^a)$

$\forall_{abcde}(a - \text{rational} \ \& \ \neg(\text{числитель}(a) - \text{even}) \ \& \ \neg(\text{знаменатель}(a) - \text{even}) \rightarrow -e/((b - c)^a d) = e/((c-b)^a d))$

Наконец, приведем прием занесения минуса под разность, применяемый в любых ситуациях, а не только при редактировании:

$\forall_{abc}(\sqrt{-a(b-c)} = \sqrt{a(c-b)})$

Уровень срабатывания этого приема равен 0.

Усмотрение противоречивого равенства с минусом

$\forall_{ab}(0 < a \rightarrow b^2 = -a \leftrightarrow \text{ложь})$

Прием используется в задачах на исследование для контроля нереализуемых подслучаев. Большое число приемов такого типа создано в геометрии. На наличие внешнего разбора случаев указывает фильтр "цель(контроль)". Поводом для попытки усмотрения противоречия служит обнаружение "подозрительного" равенства квадрата выражению, начинающемуся с минуса. Уровень срабатывания приема равен 1. Заметим, что приемы отсечения нереализуемых подслучаев должны иметь уровень срабатывания не выше 2 - иначе подслучай попадет в накопитель ответа.

Минус в обеих частях равенства

$\forall_{ab}(-a = -b \leftrightarrow a = b)$

Уровень срабатывания приема равен 0.

Равенство неотрицательного и неположительного выражений

$\forall_{ab}(0 \leq a \ \& \ 0 \leq b \ \& \ a = -b \rightarrow a = 0 \ \& \ b = 0)$

Прием выполняет вывод в посылках задачи на исследование, имеющей цель "известно" (такие задачи типичны для геометрии, физики, химии и других предметных областей, где преобладают процессы вывода следствий). Третий антецедент берется в посылках; первые два обрабатываются проверочными операторами. Попытки их обработки - очень поверхностные, из-за сильных ограничителей трудоемкости "Обрыв(1 10000)", "Обрыв(2 10000)".

Равенство вида $A = -A$

$\forall_a(a - \text{число} \to a = -a \leftrightarrow a = 0)$

Прием срабатывает на уровне 0.

Минус перед минус-бесконечностью

Так как минус-бесконечность представляется специальным логическим символом "минусбеск", для устранения минуса перед ней нужен отдельный прием (в формульной записи символ "минусбеск" прорисовывается как $-\infty$):

$- - \infty = \infty$

Уровень срабатывания нулевой.

Усмотрение равенства противоположной разности константе

$\forall_{abc}(a - b = c \to b - a = -c)$

Прием имеет заголовок "второйтерм". Для текущего рассматриваемого выражения $b - a$, имеющего слагаемые со знаком "минус", в контексте ищется равенство противоположной разности $a - b$ константе c. После этого выражение $b - a$ заменяется на $-c$. Прием срабатывает на уровне 4.

Нормализатор общей стандартизации "нормминус"

Нормализатор выполняет простейшие действия: отбрасывает минус перед нулем; удаляет двойные минусы; вносит минус под знак разности; преобразует символы плюс-минус бесконечности, предваряемые минусом. Кроме того, в нем имеется прием, отбрасывающий минус перед "О большим", используемый при вычислении пределов.

Нормализатор сокращенной записи "упрощминус"

Нормализатор относится к серии нормализаторов, используемых процедурой "свертка" при завершающей компактной переформулировке ответа. Названия таких нормализаторов, соответствующих различным заголовкам преобразуемых выражений, определяются справочником "нормупростить". Все эти названия начинаются с приставки "упрощ".

Нормализатор "упрощминус" имеет приемы, вносящие минус под знак разности - основания степени числителя либо знаменателя дроби, перед которой находится минус. Такие приемы (для сканирования задачи) выше уже рассматривались. Кроме того, имеются приемы, вносящие минус под знак разности, расположенной под нечетной тригонометрической функцией.

Нормализатор "уравнминус"

Нормализатор относится к серии нормализаторов выражений с неизвестными, применяемых процедурой "нормуравн". Она часто используется в указателях нормализации приемов ГЕНОЛОГа, относящихся к уравнениям и неравенствам элементарной алгебры. Названия нормализаторов данного типа определяются по заголовку преобразуемого выражения справочником "нормнеизв".

Нормализатор "уравнминус" имеет единственный прием, основанный на следующей теореме:

$\forall_{ab}(b = a \rightarrow -a = -b)$

Этот прием всего лишь переадресует процедуре "нормуравн" (к ней обращается антецедент) обработку выражения a, расположенного под минусом. Необходимо наличие комментария "нормуравн", определяющего сквозную обработку подвыражений.

10.5 Приемы символа "плюс"

Выражение "плюс($a_1 \dots a_n$)", прорисовываемое формульным редактором в виде $a_1 + \dots + a_n$, обозначает сумму чисел $a_1, \dots, a_n$. Начнем с перечня приемов справочников, созданных для символа "плюс". Справочник "тип" указывает числовой тип значений операции; справочники "ассоциативно" и "коммутативно" - что операция ассоциативная и коммутативная. Справочник "арность" дает значение арности 2, однако в сочетании со свойством ассоциативности это позволяет рассматривать операцию от любого числа операндов, большего единицы. Справочник "одз" имеет теорему приема вида "длялюбого(x1 x2 эквивалентно(определено(плюс x1 x2)и(число(x1)число(x2))))". Хотя в ней рассматривается двуместный случай, компилятор создает программу приема для обработки суммы любого числа слагаемых. Она находит все слагаемые, для которых из контекста не усматривается их числовой тип (применяется оператор "усмчисло"), и выдает конъюнкцию утверждений о том, что значения слагаемых - числа. Справочник "типданных" имеет такую же теорему приема. Справочник "суммавсех" указывает, что имеется "векторный" аналог суммы, применяемый к конечным числовым семействам - операция "суммавсех". Справочник "область" говорит о том, что значением суммы может быть произвольное число. Справочник "единица" определяет единицу операции - логический символ "0". Справочник "монотонно" указывает, что значение операции монотонно неубывает при монотонном неубывании операнда. Справочник "вычисл" используется компилятором ГЕНОЛОГа. Он предлагает операторные выражения для вычисления суммы, в зависимости от типа данных слагаемых. Один из приемов этого справочника формирует оператор для решения квадратного уравнения с численными коэффициентами. Справочник "степень" сопоставляет операции n - кратного сложения числа с самим собой операцию умножения этого числа на n. Справочник "знаксуммы" указывает на операцию "минус" как допускающую внесение под знак суммы. Справочник "внутрзамена" играет аналогичную роль, но его входными данными служат оба символа "плюс" и "минус", и он только проверяет возможность внесения одноместной операции под двуместную. Справочники "группировки", "разделение", "группировка", "заголовок", "неизвоценка" используются при выводе в базе теорем и автоматическом синтезе приемов.

Простейшая стандартизация

Для устранения вложенных сумм служит прием с заголовком "спускоперандов", базирующийся на теореме "коммутативно(плюс)". На этой же теореме основан прием лексикографического упорядочения слагаемых, имеющий заголовок "лексупорядочение". Уровни срабатывания обоих приемов равны 0. Для устранения нулевого слагаемого имеется прием $\forall_a(a + 0 = a)$. Он применяется на нулевом уровне без ограничений, даже в сопровождающих по о.д.з. утверждениях. Указатель "нормализатор" инициирует дополнительное упрощение надтермов.

Сложение десятичных слагаемых

$\forall_{abcd}(c = a + b \rightarrow a + b + d = c + d)$

a, b идентифицируются с десятичными записями. Антецедент использует нормализатор "сложить". Указатель "нормализатор" инициирует попытки упрощения надтермов. Прием срабатывает на уровне 0.

Сложение простых дробей

Если в сумме встречаются две простые дроби либо простая дробь и десятичная запись, то они складываются. Исключение составляет случай смешанных дробей - суммы целого числа и простой дроби того же знака. Теорема приема имеет такой же вид, как в предыдущем пункте. Заметим, что если надоперацией суммы являются умножение, дробь либо степень, то складываются даже компоненты смешанной дроби. Однако, этого не происходит, если внешнее умножение имеет своим сомножителем единицу измерения. Для сложения используется нормализатор "сложитьдроби".

Вынесение за скобку общего минуса всех слагаемых

Общий минус всех слагаемых выносится за скобку только при условии, что усматривается возможность дальнейших преобразований, использующих этот минус. Прием срабатывает на уровне 0. Его теорема имеет вид:

$\forall_{ab}(-a - b = -(a + b))$

Указатель "набор(второйтерм)" дает возможность обрабатывать любое число слагаемых, не обязательно равное двум. Преобразование выполняется в следующих случаях:

1. Сумма служит операндом одного из следующих символов: "делит", "целое", "числитель", "знаменатель", "рациональное", "минус", "умножение", "дробь", "модуль", "сигнум", "синус", "косинус", "тангенс", "секанс", "косеканс", "арксинус", "арккосинус", "арктангенс", "арккотангенс", "Умножение".
2. Сумма является первым операндом операции "умножвект".
3. Сумма расположена под квадратным радикалом, умножаемым на мнимую единицу.
4. Сумма является операндом одного из отношений "равно", "меньше", "меньшеилиравно". Либо противоположный операнд имеет знак минус, либо он равен 0, либо сумма содержит неизвестные, а противоположный операнд не содержит, либо сумма неконстантная, а противоположный операнд константный.
5. Сумма является основанием степени с рациональным показателем, имеющим нечетный знаменатель.

Дополнительно введен ограничитель, блокирующий прием в тех случаях, когда может быть нарушена специальная стандартизация неравенств для вещественных и мнимых частей, описывающих области на комплексной плоскости. У этих неравенств ноль должен находиться в левой части. При вынесении минуса наружу произойдет перестановка частей неравенства, и стандартизация нарушится.

Приведение подобных членов с числовыми коэффициентами

Для приведения подобных членов с числовыми коэффициентами созданы несколько приемов. Начнем с простейшего и наиболее часто употребляемого:

$\forall_{abcd}(d = b + c \to ab + ac = da)$

Переменные b, c идентифицируются с десятичными записями. Если число слагаемых велико (больше 15), то прием срабатывает на уровне 1, иначе - на уровне 0. Небольшая задержка срабатывания нужна для того, чтобы успеть привести к стандартному виду одночлены, появившиеся после раскрывания скобок - перемножить степени с одинаковыми основаниями, вынести наружу знаки, и т.п. Приведение подобных членов для необработанных слагаемых дает большое число холостых попыток и, в случае длинной суммы, может ощутимо замедлить решение. Впрочем, обычно раскрывание скобок осуществляется пакетным нормализатором "стандплюс", который одновременно приводит подобные члены. На долю приемов сканирования задачи остается лишь небольшая часть этой работы.

Чтобы не дублировать рассмотрение пар слагаемых, введен фильтр "постпозиция(фикс(0 1 2)фикс(0 1 1))". В результате с выражением ac будут идентифицироваться только те слагаемые, которые расположены в сумме правее слагаемого ab. Указатели "наборoперандов(фикс(0 1 1))", "наборoперандов(фикс(0 1 2))" определяют режим выделения общей части a двух рассматриваемых слагаемых без участия трудоемкого оператора "алгебрпересечение": она определяется как обычное пересечение списков сомножителей двух слагаемых. Указатели "единица(1 x2 x3)", "заменазнака(минус x2 x3)" разрешают коэффициентам b, c обращаться в единицу и относят к ним знаки "минус" перед слагаемыми. Антцедент, складывающий коэффициенты, выделен указателем "программа".

Следующие три приема приведения подобных членов относятся к случаям дробных слагаемых:

$$\forall_{abcde}\left(e = a + cd \to \frac{ab}{c} + db = \frac{eb}{c}\right)$$

$$\forall_{abdefpq}\left(p = bf + de \;\&\; q = df \to \frac{ab}{d} + \frac{ae}{f} = \frac{pa}{q}\right)$$

$$\forall_{abcdefpq}\left(p = bf + de \;\&\; q = df \to \frac{ab}{cd} + \frac{ae}{cf} = \frac{pa}{qc}\right)$$

В первом случае одно слагаемое дробное, другое - нет. В последнем случае имеется неединичая общая часть c двух знаменателей, а во втором - ее нет. Уровни срабатывания всех трех приемов равны 0, в остальном они аналогичны приведенному выше приему. Если третий прием снабдить указателем "единица(1 c)", то второй прием будет не нужен, но произойдет небольшое замедление идентификации.

Кроме приведения подобных членов, коэффициентами которых служат десятичные записи, введены два приема для случая константных коэффициентов произвольного вида. Они имеют следующую теорему:

$\forall_{abc}(ab + ac = a(b + c))$.

Здесь b, c идентифицируются с константными выражениями, a - с неконстантным выражением. Один прием используется при решении задачи на исследование, имеющей цель "известно", другой - при редактировании ответа такой задачи. Уровни срабатывания равны, соответственно, 2 и 3.

Приведение подобных членов с неизвестными

Если сумма содержит неизвестные, то при приведении подобных членов коэффициент формируется из всех известных сомножителей одночлена. Теорема приема такая же, как выше:

$\forall_{abc}(ab + ac = a(b + c))$

Выражения b, c не содержат неизвестных, a - содержит. Текущая задача - на описание, исследование либо на преобразование. В последнем случае речь идет о вспомогательных преобразованиях подвыражений с неизвестными, и задаче на преобразование передается цель (неизвестные ...) из внешней задачи. Указатели "набороперандов(фикс(0 1 1))", "набороперандов(фикс(0 1 2))" определяют идентификацию a по пересечению множеств сомножителей двух слагаемых. Уровень срабатывания приема равен 1.

Если задача имеет цель (независит $y_1 \dots y_m$), причем хотя бы один из коэффициентов b, c содержит какую-либо переменную $y_i, i \in \{1, \dots, m\}$, то прием блокируется. Это делается потому, что для устранения зависимости ответа от y_i используется альтернативная группировка слагаемых в член вида $(t_1 + \dots t_k)F(y_i)$. После группировки не содержащая y_i сумма $t_1 + \dots t_k$ приравнивается нулю, и таким образом зависимость снимается.

Прием блокируется также для сумм, расположенных под знаком тригонометрической операции, так как здесь удобнее представлять аргумент в виде суммы. Наконец, прием блокируется при наличии внешнего квантора либо описателя, связывающего переменные суммы.

Дробные коэффициенты в уравнениях и неравенствах обычно исключаются другими приемами. Однако, для ускорения вычислений был введен прием, приводящий неизвестные подобные члены дробного вида:

$$\forall_{abcdx}\left(\frac{ax}{b} + \frac{cx}{d} = \left(\frac{a}{b} + \frac{c}{d}\right)x\right)$$

Переменные a, b, c, d идентифицируются с выражениями без неизвестных, x - с выражением, имеющим неизвестные. Если задача имеет цель "областьграницы", то уровень срабатывания равен 0, иначе он равен 2. Такая цель появляется лишь в задачах на кратные интегралы при определении плоской области, заданной своими граничными линиями.

Попытка вынесения общего множителя всех слагаемых

Попытка предпринимается в задаче на преобразование, если рассматриваемая сумма корневая. Теорема приема - такая же, как при приведении подобных членов, но имеется указатель "набор(второйтерм)", определяющий обработку сразу всех слагаемых. Созданы две версии приема, срабатывающие на уровне 1. Одна из них определяет общий множитель по пересечению групп сомножителей. Другая - использует более сильный оператор "алгебрпересечение", позволяющий извлекать неявные общие множители, однако она применяется лишь в задачах на упрощение производной. В обоих случаях имеются дополнительные ограничители срабатывания, учитывающие специальные целевые установки (например, раскрывание скобок).

Внесение минуса под знак суммы

Если сумма, заключенная под знаком "минус", является слагаемым внешней суммы, то знак "минус" вносится под сумму.

$\forall_{abc}(c - (a + b) = c - a - b)$

Правая часть обрабатывается нормализатором "нормплюс", который имеет такой же прием. Поэтому знак "минус" оказывается отнесенным к отдельным слагаемым суммы $a + b$, даже если их число было более двух. Уровень срабатывания приема равен 0.

Группировка всех ненулевых слагаемых в одной части равенства

За исключением особых случаев, все ненулевые слагаемые находящегося в условии задачи числового равенства без неизвестных группируются в левой части. Теорема приема имеет вид:

$\forall_{ab}(a - \text{число} \ \& \ b - \text{число} \rightarrow a = b \leftrightarrow a - b = 0)$

Прием срабатывает на уровне 0; проверяется, что выражения a, b не являются нулями. Таким образом, наличие нуля в левой части равенства не приводит к срабатыванию данного приема, однако если прием срабатывает, то нуль оказывается в правой части. Для срабатывания приема необходимо, чтобы равенство либо располагалось под отрицанием, являющимся посылкой или условием (такие отрицания называем "корневыми"), либо находилось в условии задачи, либо представляло собой посылку задачи на исследование. Приведем ряд дополнительных ограничений на срабатывание приема:

1. Выполнено хотя бы одно из условий:
    (a) Задача имеет тип "доказать";
    (b) Равенство не содержит неизвестных. Если происходит редактирование ответа задачи на описание либо равенство не расположено под корневым отрицанием, то оно не имеет вида $x = t$, где x - переменная, не входящая в t.
    (c) Равенство расположено под логическим символом, отличным от символов "существует", "длялюбого", "и", "или", "не".
2. Если редактируется ответ задачи на описание, то отсутствует комментарий "стандменьше", указывающий, что уже начато разрешение сопровождающих условий относительно известных параметров.
3. Текущая задача на описание не имеет цели (известно ...).
4. Текущая задача на исследование не связана с семантическим анализом фразы.
5. Ни одно из выражений a, b не является переменной, связанной внешним описателем либо квантором существования.
6. Если не рассматривается условие задачи на доказательство, в котором отсутствуют связанные внешними кванторами и описателями переменные, то ни одна из частей равенства не является отличным от переменной числовым атомом, не

входящим в противоположную часть. Напомним, что числовыми атомами мы условились называть числовые переменные, а также такие числовые выражения $f(t_1 \ldots t_n)$, у которых хотя бы одно t_i имеет нечисловое значение. Примеры невырожденных числовых атомов - "масса(A)", "расстояние(AB)", и т.п.

7. Если равенство не имеет свободных переменных и не является условием задачи на доказательство, то ни одна из его частей не является числовым атомом. Кроме того, в этой ситуации обе части одновременно либо имеют, либо не имеют числовых атомов.

Равенство нулю суммы слагаемых одного знака

Если все слагаемые имеют один и тот же нестрогий знак, а сумма их равна нулю, то каждое слагаемое равно 0. Этот прием встречается в искусственно составленных задчах. Однако, попытки применять его без ограничений к любым равенствам суммы нулю могли бы привести к неоправданно большим затратам трудоемкости. Поэтому созданы несколько приемов такого типа, ориентированных на специальные случаи.

$\forall_{ab}(a \leq 0 \ \& \ b \leq 0 \to a+b=0 \leftrightarrow a=0 \ \& \ b=0)$

Прием применяется к условию задачи на описание. Преобразуемое равенство - корневое либо находится под корневым отрицанием. Слагаемое a имеет своим сомножителем степенное выражение. Уровень срабатывания, по сравнению с другими приемами раздела, повышен (равен 5), так как ситуация со всеми неположителными слагаемыми менее вероятна, чем ситуация со всеми неотрицательными. Например, прием, выносящий наружу общий минус, преобразует первую во вторую.

$\forall_{ab}(0 \leq a \ \& \ 0 \leq b \to a+b=0 \leftrightarrow a=0 \ \& \ b=0)$

Прием аналогичен предыдущему, но уровень срабатывания равен 2. Требуется, чтобы слагаемое a имело степенной множитель с неконстантным основанием. Прием не применяется при наличии цели "стандравно". Такая цель имеется у вспомогательных задач на описание, разрешающих группу посылок задачи на исследование. Например, она часто возникает в геометрии. В этих ситуациях срабатывание приема маловероятно. Прием блокируется, если равенство находилось под корневым отрицанием, а заменяющий терм содержит дизъюнкцию. Заметим, что дизъюнкция может появиться в результате работы нормализатора "нормчисло", обрабатывающего оба результирующих равенства.

На предыдущей теореме основаны еще два приема. В первом из них слагаемое a должно иметь своим сомножителем модуль некоторого выражения. Во втором - все слагаемые имеют знак "плюс", причем a имеет своим сомножителем степень с неконстантным основанием. Оба приема срабатывают на уровне 0. Второй из них имеет сильные ограничители трудоемкости для проверок неравенств. Создан аналог второго приема для неположительных слагаемых. Он тоже срабатывает на уровне 0. Наконец, приведем еще два приема, ориентированных на частные случаи:

$\forall_{abcd}(0 < a \ \& \ 0 < c \to a\sqrt{b} + c\sqrt{d} = 0 \leftrightarrow b=0 \ \& \ d=0)$

$\forall_{ab}(a^2+b^2=0 \leftrightarrow a=0 \ \& \ b=0)$

Оба приема применяются в задачах на исследование. В первом из них a, c идентифицируются с константами. Для применения второго приема необходимо, чтобы либо равенство находилось вне кванторов и описателей, связывающих его переменные, либо являлось операндом конъюнкции или описателя "класс". Первый прием срабатывает на уровне 3, второй - на уровне 1.

Группировка в одной части всех слагаемых уравнения

Неизвестные слагаемые уравнения группируются решателем в левой части, известные - в правой. Однако, это достигается совместной работой нескольких приемов. Первый из них, обнаружив неизвестные в обеих частях равенства, переносит все ненулевые члены влево:

$\forall_{ab}(a - \text{число} \to a = b \leftrightarrow a - b = 0)$

Прием применяется к условию задачи на описание либо к посылке задачи на исследование, не имеющей цели "известно". Рассматриваемое равенство может располагаться только под символами "и", "или", "существует". Оба выражения a, b содержат неизвестные. Прием блокируется при выполнении хотя бы одного из следующих условий:

1. Происходит редактирование ответа задачи на описание.
2. Одно из выражений a, b - неизвестная, не входящая в другое.
3. Одно из выражений a, b имеет вид $y(x)$, где x, y - переменные; y - неизвестная, не входящая в противоположное выражение; переменная x входит в цель (связка $\ldots$). Заметим, что цель "связка" появляется при решении функциональных уравнений и перечисляет варьируемые переменные. Подробнее о ней будет говориться в разделе, посвященном дифференциальным уравнениям.
4. Одно из выражений имеет заголовок "расстояние" либо "угол".
5. Решается задача на исследование, причем одно из выражений a, b не входит в другое и имеет вид $y(t)$, где y - неизвестная; t - выражение без неизвестных.

Прием срабатывает на уровне 1.

Другой случай группировки всех ненулевых членов уравнения в одной части - блокировка нецелесообразной подстановки вместо неизвестной x неизвестного выражения t в остальные уравнения задачи. Такая подстановка была бы выполнена на уровне 1 согласно уравнению $x = t$, и лишь в особых случаях оказалсь бы полезной. Поэтому на уровне 0 применяется прием, имеющий следующую теорему:

$\forall_{abc}(a = b + c \leftrightarrow a - b - c = 0)$.

Уравнение является условием задачи на описание, число неизвестных которой не менее двух. a - неизвестная, не входящая в правую часть. Последняя нелинейна относительно хотя бы двух неизвестных. На этапе редактирования ответа прием блокируется.

Решение уравнения $X + A = B$

Фактически этот прием выполняет группировку неизвестных слагаемых в левой части уравнения, а известных - в правой:

$\forall_{abc}(c - \text{число} \to a + b = c \leftrightarrow a = c - b)$

Указатель "перечень(x1 не(известно(x1)))" определяет идентификацию a с суммой всех неизвестных слагаемых одной из частей уравнения. Другая часть c не содержит неизвестных. Остаток b непустой; сумма a невырожденная. Уравнение находится в условии задачи на описание либо в посылке задачи на исследование. Оно может располагаться только под символами "и", "или", "существует". Если уравнение расположено под квантором существования, то этот квантор должен представлять собой параметрическое описание серии решений либо допускать несложное преобразование к такому виду. В обычной ситуации уровень срабатывания приема равен 1, однако в двух специальных случаях он понижен до нуля (уравнение под дизъюнкцией либо текущая задача на исследование имеет цель "известно"). При редактировании ответа прием блокируется.

На указанной выше теореме основаны еще несколько различных приемов:

1. При редактировании ответа задачи на описание преобразуется равенство без неизвестных. Выражение c константное; b идентифицируется с суммой всех неконстантных слагаемых и невырождено. Остаток a непустой. Применение приема сопровождается вводом комментария "стандменьше", указывающим на этап разрешения известных утверждений ответа относительно параметров. Если такой комментарий уже имелся, то уровень срабатывания равен 1, иначе он равен 5. Равенство не расположено под квантором либо описателем, связывающим его переменные.

2. Преобразуется посылка задачи на доказательство, имеющей комментарий (известно ...), выделяющий часть переменных как "известные". Такие комментарии вводятся при решении геометрических задач. a идентифицируется с суммой всех "не известных" слагаемых; c известно. Прием срабатывает на уровне 0.

3. Преобразуется посылка задачи на исследование, имеющей цель "контроль", т.е. возникшей при разборе случаев. a - неизвестная, не входящая в b; c известно. Требуется наличие посылки, имеющей символ "коорд" (координаты точки в некоторой аффинной системе координат) и пересекающейся с рассматриваемым равенством по своим свободным переменным. Прием обеспечивает подстановку вместо неизвестной a ее выражения $c - b$. Таким образом увеличивается вероятность обнаружения противоречивого подслучая. Уровень срабатывания равен 2.

4. Равенство представляет собой посылку задачи на исследование. a - переменная; b, c - константные выражения. Уровень срабатывания равен 2.

Исключение неизвестной X с помощью уравнения $X + A = B$

В определенных случаях бывает целесообразно выразить одну из неизвестных через другие для последующего ее исключения. Один из простейших приемов такого типа для числовой неизвестной основан на следующей теореме:

$\forall_{abc}(a - \text{число} \ \& \ b - \text{число} \rightarrow a = b + c \leftrightarrow b = a - c)$

Преобразуемое равенство - условие задачи на описание либо посылка задачи на исследование. a не содержит неизвестных; c содержит неизвестные; b - неизвестная, не входящая в c. В задачах на описание прием срабатывает на уровнях 1 либо 5; в задачах на исследование - на уровнях 0 и 5. Дополнительно накладываются следующие ограничения:

1. Либо уравнение линейно по всем своим неизвестным, либо a есть 0, а число неизвестных в уравнении не более двух, либо решается задача на описание, а текущий уровень равен 5.

2. Либо уравнение имеет не более двух неизвестных, либо решается задача на описание, имеющая не более одного нелинейного уравнения, либо решается задача на исследование, a равно 0, а число неизвестных в уравнении равно 3.
3. Либо неизвестная b не является целочисленной, либо решается задача на исследование, либо число неизвестных, входящих в c, не более двух.
4. Отсутствует цель "независит ...", указывающая на независимость ответа от какой-либо переменной, входящей в a либо в c.

Таким образом, в основном прием применяется к системам уравнений, близким к линейным, либо к уравнениям с двумя неизвестными.

Исключение неизвестной X с помощью уравнения $A - X = B$

Прием аналогичен предыдущему:

$\forall_{abc}(c - \text{число} \ \&\ b - \text{число} \to a - c = b \leftrightarrow c = a - b)$

c - неизвестная; a не известно и не содержит c; b известно.

Исключение неизвестной X с помощью уравнения $AX + B = C$

Этот прием обобщает два предыдущих, однако его уровни срабатывания больше. Таким образом достигается первоочередное исключение неизвестных, не требующее деления.

$$\forall_{abcd}\left(\neg(a = 0) \to ad + b = c \leftrightarrow d = \frac{c - b}{a}\right)$$

Вместо уровней срабатывания 0,1,5 берутся уровни 1,2,6 соответственно. d - неизвестная, b не известно и не содержит d, a, c известны. Прочие условия на срабатывание - такие же, как выше.

Вычитание уравнения, домноженного на известный коэффициент

Если в двух уравнениях имеются два члена с одной и той же неизвестной x, отличающиеся лишь известными сомножителями, а в других членах x не встречается, то одно из уравнений преобразуется для исключения x:

$\forall_{abcdefgx}(agx + b = c \ \&\ \neg(a = 0) \to dgx + e = f \leftrightarrow ae - bd = af - cd)$

Преобразуемое уравнение является условием задачи на описание, имеющей более одной неизвестной. x - неизвестная; a, d, c, f известны; x не входит в b, e. Каждая неизвестная выражения b встречается в e, т.е. результирующее уравнение не приобретает новых неизвестных. Либо преобразуемое уравнение не линейно относительно какой-либо из своих неизвестных, либо оба уравнения линейны. Задача не имеет целей (известно ...), (независит ...). Первый антецедент идентифицируется с дополнительным уравнением задачи, второй - обрабатывается проверочным оператором. Прием срабатывает на уровне 1.

Следующий прием выполняет аналогичное преобразование в более общей ситуации:

$\forall_{abcdefgh}(\neg(g = 0) \ \&\ ag + b = c \ \&\ f = bh - dg - ch + eg \to ah + d = e \leftrightarrow f = 0)$

Уравнение является условием задачи на описание, имеющей несколько неизвестных. При вычитании исключаются "подобные" члены ag и ah. Выражение a не известно; g идентифицируется с произведением всех известных множителей первого члена; h известно. Второй антецедент идентифицируется с дополнительным условием задачи; третий антецедент, выделенный указателем "идентификатор", определяет левую часть f линейной комбинации уравнений. Эта часть, в которой сгруппированы все ненулевые члены, обрабатывается после общей стандартизации нормализатором "факторизация". При разложении ее на несколько неизвестных множителей результирующее уравнение, обрабатываемое нормализатором "нормчисло", распадется в дизъюнкцию более простых уравнений. Для срабатывания приема необходимо выполнение ряда дополнительных требований:

1. Не имеет места этап редактирования ответа.
2. Рассматриваемые уравнения не имеют дробных слагаемых.
3. Либо выражение f имеет единственную неизвестную, а преобразуемое уравнение - более одной неизвестной, либо заменяющее утверждение имеет заголовок "или", либо заменяющее утверждение представляет собой линейное уравнение, либо в f меньше слагаемых, имеющих сомножителем дробную степень с неизвестным основанием, чем в левой части преобразуемого уравнения.
4. Задача не имеет целей "известно", "независит".
5. Преобразуемое уравнение, а также уравнение, полученное из него изменением знаков всех членов левой и правой частей, не входят в заменяющий терм.
6. Либо число неизвестных, входящих в выражения b, d, равно 1, либо каждое из этих выражений линейно относительно своих неизвестных, либо b, d имеют еще пару подобных неизвестных слагаемых с теми же известными коэффициентами g, h, либо a имеет сомножителем дробную степень с неизвестным основанием, а b, d не имеют слагаемых с таким сомножителем.
7. Если преобразуемое и результирующее уравнения линейны относительно всех своих неизвестных, то число неизвестных результирующего уравнения меньше, чем у преобразуемого.

Указатель "набороперандов(фикс(2 1 1)фикс(0 1 1 1))" определяет идентификацию общей части a двух слагаемых с помощью обычного пересечения списков сомножителей, без трудоемкой процедуры "алгебрпересечение". Уровень срабатывания приема равен 2.

Выражение одной неизвестной через другие, если уравнение относительно этой неизвестной линейно

Прием отличается от рассмотренного выше приема исключения неизвестной X из уравнения $AX + B = C$ тем, что коэффициент A может содержать неизвестные и X может входить в B. Он применяется только в задачах на исследование. Разрешение относительно неизвестной выполняется вспомогательной задачей, так что теорема приема имеет вид обращения к задаче:

$\forall_{abcd}(ab + c = d \leftrightarrow ab + c = d)$

Текущая задача на исследование не имеет цели "известно". b - неизвестная, не входящая в a. Допускаются вырожденные случаи равенства a единице либо c нулю, но не оба одновременно. Прием срабатывает на уровнях 2 и 6. Если текущий уровень равен 2, то правая часть d есть 0, a известно, c имеет вид произведения известного множителя на неизвестную, причем в задаче существует уравнение, не линейное относительно b. Если текущий уровень равен 6, то требуется лишь, чтобы c было линейно относительно b. Заменяющий терм обрабатывается задачей на описание, имеющей единственную неизвестную b. Дополнительные ограничения, необходимые для срабатывания приема, таковы:

1. Если рассматриваемое уравнение линейно относительно своих неизвестных, то оно самое короткое из таких уравнений. Если оно нелинейно, то задача не имеет линейных уравнений и не имеет более короткого уравнения, линейного относительно какой-либо неизвестной.
2. Неизвестная b не встречается в уравнениях внутри степеней с дробными показателями.
3. Пусть текущий уровень равен 6; имеется уравнение, не линейное относительно b, причем либо a не известно, либо b встречается в c. Тогда отсутствует такая неизвестная задачи, относительно которой линейны c и все содержащие ее уравнения. Это условие позволяет выбирать для разрешения "наилучшую" из неизвестных текущего уравнения.
4. Если неизвестная b целочисленная, то она в уравнении единственная.
5. Если выражение a содержит неизвестные, то c не представимо в виде $py+q$, где y - отличная от b неизвестная, не входящая в a, q; p известно.

Усмотрение общего неизвестного множителя в двух уравнениях с нулевой правой частью

Если два уравнения задачи на описание имеют суммы в левой части и нули в правой, то предпринимается попытка разложить левые части на множители и усмотреть их общий неизвестный множитель:

$\forall_{abcde}(ab = c \;\&\; ad = e \rightarrow c = 0 \;\&\; e = 0 \leftrightarrow a = 0 \;\vee\; \neg(a = 0) \;\&\; b = 0 \;\&\; d = 0)$

Прием имеет заголовок "заменаусловия(второйтерм)" и выполняет одновременную замену двух уравнений. Антецеденты обращаются к нормализатору "факторизация" для быстрой попытки разложения на множители сумм c и e. Определяются общий множитель a результатов разложения и остаточные произведения b, d. Проверяется, что a содержит неизвестные и хотя бы одно из выражений b, d тоже содержит неизвестные. Прием применяется на уровне 3 при условии, что задача имеет более одной неизвестной.

Система из двух уравнений, линейных относительно двух выражений с неизвестной, не встречающейся вне этих выражений

Если в каждом из двух уравнений встречается линейная комбинация двух неизвестных выражений x, y, сопровождаемая, быть может, прочими неизвестными слагаемыми, то можно разрешить эти уравнения относительно x и y. В полученных уравнениях x и y окажутся разделены. Если, к тому же, некоторая неизвестная в рассматриваемых уравнениях встречается только внутри x либо y, то дальнейшее решение может существенно упроститься.

$\forall_{abcdefghp}(p = af - be \rightarrow ax + by + c = d \;\&\; ex + fy + g = h \leftrightarrow \neg(p = 0) \;\&\; px - (d - c)f + (h - g)b = 0 \;\&\; py - (h - g)a + (d - c)e = 0 \;\vee\; p = 0 \;\&\; ax + by + c = d \;\&\; ex + fy + g = h)$

Прием имеет заголовок "заменаусловия(второйтерм)" и выполняет одновременную замену двух уравнений $ax+by+c = d, ex+fy+g = h$. Он применяется в задаче на описание, имеющей более одной неизвестной. Указатель "контекст(неизвестная(x9))" выбирает некоторую неизвестную x9. Указатели "перечень(x1 не(входит(x9 x1)))", "перечень(x2 не(входит(x9 x2)))" определяют идентификацию коэффициентов a, b с произведениями всех множителей, не содержащих выделенной неизвестной. Аналогичные указатели определяют идентификацию коэффициентов e, f. Проверяется, что неизвестная x9 входит как в x, так и в y, не совпадая с ними, и что она не встречается в выражениях c, d, g, h. Антецедент вычисляет определитель p линейной системы относительно x, y, используя нормализаторы общей стандартизации и предпринимая попытку разложения результата на множители. Проверяется, что после упрощений p не оказалось тождественно нулевым. Результирующие уравнения обрабатываются различными нормализаторами; в частности, предпринимается быстрая попытка разложения их левых частей на множители. Уровень срабатывания приема равен 4.

Попытка факторизации разности либо суммы уравнений

Иногда составители задач берут какую-либо сумму, допускающую разложение на множители, разбивают ее на две части, приравнивают их нулю и получают систему уравнений, для решения которой нужно догадаться сложить либо вычесть уравнения. Против таких задач направлены следующие два приема:

$\forall_{abcde}(e = a - b + c - d \;\&\; a = b \rightarrow c = d \leftrightarrow e = 0)$

$\forall_{abcde}(e = a - b + d - c \;\&\; a = b \rightarrow c = d \leftrightarrow e = 0)$

Первый из приемов складывает уравнения, второй - вычитает. Антецедент обращается к нормализатору "видумножение", обеспечивающему попытку разложения на множители с помощью практически всех имеющихся у решателя средств. Проверяется, что получаемое выражение e имеет хотя бы два неизвестных сомножителя либо является неизвестной степенью. Приемы применяются на уровне 7 в задаче на описание, имеющей более одной неизвестной. Введен ограничитель трудоемкости, однако не очень сильный.

Сложение либо вычитание уравнений для получения в левой части степени суммы

Сложение либо вычитание двух уравнений с известными правыми частями может оказаться полезным, если левая часть результата представима в виде степени с неизвестным основанием. Извлекая корень, получим явное выражение для этого основания.

$\forall_{abcde}(e = a + c \;\&\; a = b \rightarrow c = d \leftrightarrow e = b + d)$

$\forall_{abcde}(e = a - c \;\&\; a = b \rightarrow c = d \leftrightarrow e = b - d)$

Первый прием складывает уравнения, второй - вычитает. Выражения a, c содержат неизвестные, выражения b, d - не содержат. Приемы срабатывают на уровне 6 в задачах на описание, имеющих более одной неизвестной. Проверяется, что e имеет вид произведения известного выражения на степень с неизвестным основанием. К уравнениям с неизвестными дробными степенями приемы не применяются. Приемы блокируются при редактировании ответа, а также при наличии цели "известно". Заметим, что для разложения на множители приемы используют ослабленный нормализатор "факторизация".

Система из двух линейных уравнений с двумя неизвестными, имеющая численные коэффициенты

Для ускоренного рассмотрения простейших линейных систем с двумя неизвестными, имеющих константные коэффициенты (не обязательно десятичные записи), создан специальный прием, срабатывающий на уровне 0:

$\forall_{abcdefpqrxy}(p = ae - bd \;\&\; q = ce - bf \;\&\; r = af - cd \;\&\; \neg(p = 0) \rightarrow ax + by = c \;\&\; dx + ey = f \leftrightarrow x = q/p \;\&\; y = r/p)$

Переменные x, y идентифицируются с неизвестными; a, b, c, d, e, f - с константными выражениями. Первые три антецедента, выделенные указателем "идентификатор", обрабатывают свои правые части нормализаторами общей стандартизации, после чего предпринимают попытки разложить их на множители нормализатором "видумножение".

Усмотрение подсистемы из трех линейных уравнений относительно трех нелинейных неизвестных выражений

Если усматривается подсистема из трех линейных уравнений относительно трех нелинейных неизвестных выражений x, y, z, то предпринимается попытка разрешить ее относительно вспомогательных неизвестных x', y', z', временно замещающих x, y, z, а затем выполнить обратное замещение. Теорема этого приема вырожденная: ее левая часть нужна для идентификации трех уравнений, а правая, совпадающая с левой и обрабатываемая нормализатором, определяет результат указанных выше действий.

$\forall abcdefghpqrstuvxyz(ax + by + cz + d = e \;\&\; fx + gy + hz + p = q \;\&\; rx + sy + tz + u = v \leftrightarrow ax + by + cz + d = e \;\&\; fx + gy + hz + p = q \;\&\; rx + sy + tz + u = v)$

Прием имеет заголовок "заменаусловия(второйтерм)". Он применяется к условиям задачи на описание, имеющей не менее трех неизвестных. Каждое из выражений x, y, z имеет неизвестную задачи, относительно которой оно нелинейно. Ни одно из выражений d, p, u не является нелинейным относительно какой-либо неизвестной. Выражения e, q, v не содержат неизвестных. Левая часть хотя бы одного из уравнений имеет более трех слагаемых. Указатели "перечень(...)" определяют идентификацию коэффициентов $a, b, c, f, g, h, r, s, t$ из всех известных сомножителей соответствующих слагаемых. Указатели "подстановка(...)" разрешают идентификацию переменных c, g, r с нулями при отсутствии соответствующих членов. Таким образом, каждое выражение x, y, z может быть пропущено не более чем в одном уравнении. Заменяющий терм обрабатывается нормализатором "задача(4 тип(описать) полный явное прямойответ цель(вспомописание(x23 x24 x25)))". Элемент "цель(вспомописание(x23 x24 x25))" означает, что будет создана вспомогательная задача на описание с новыми неизвестными x', y', z', замещающими x, y, z везде в рассматриваемой задаче, а в ее ответе данные неизвестные будут заменены на x, y, z. Прием срабатывает на уровне 3.

Линейное однородное уравнение с двумя неизвестными

Линейное однородное уравнение с двумя неизвестными позволяет свести одну неизвестную к другой, не усложняя принципиальным образом вида прочих уравнений. Поэтому исключение неизвестной из такого уравнения вынесено в отдельный прием, срабатывающий на пониженном уровне.

$$\forall_{abxy}\left(ax + by = 0 \leftrightarrow \neg(a = 0)\ \&\ x = -\frac{by}{a}\ \vee\ a = 0\ \&\ by = 0\right)$$

Прием срабатывает на уровне 3 в условии задачи на описание. Уравнение - корневое. x, y - неизвестные; a, b известны.

Ускоренный разбор случаев для дизъюнкций, члены которых линейным образом выражают одну неизвестную через другую

Если среди условий задачи на описание встречается дизъюнкция двух линейных однородных уравнений с двумя неизвестными, то вводится комментарий "разборслучаев", ускоряющий разбор случаев по данной дизъюнкции.

$\forall_{abcdxy}((ax + by = 0\ \vee\ cx + dy = 0) \to \emptyset)$

Прием имеет заголовок "замечание". Это означает, что его теорема - фиктивная. В ней используется лишь антецедент, определяющий идентификацию теоремных переменных. Указатели приема обычным образом задают необходимые действия с комментариями. Переменные x, y идентифицируются с неизвестными; a, b, c, d - с выражениями без неизвестных. Указатель "примечание(разборслучаев)" сопровождает дизъюнктивное условие, идентифицированное с антецедентом, примечанием "разборслучаев". Консеквент $\emptyset$ для теорем таких приемов является стандартом. Впрочем, компилятор его игнорирует, и на его месте можно помещать любой терм. Уровень срабатывания приема равен 0.

Однородная система из двух псевдолинейных уравнений с неизвестными коэффициентами

Если имеются два однородных уравнения относительно двух неизвестных выражений e, f, причем коэффициенты уравнений сами неизвестны, то рассматриваются два случая: либо оба выражения e, f равны 0, либо уравнения зависимы.

$\forall_{abcdefg}(\neg(a^2 + b^2 = 0)\ \&\ g = ad - bc \to ae + bf = 0\ \&\ ce + df = 0 \leftrightarrow e = 0\ \&$
$f = 0\ \vee\ ae + bf = 0\ \&\ g = 0)$

Прием применяется в условиях задачи на описание, имеющей более одной неизвестной. Оба выражения e, f содержат неизвестные, причем хотя бы одно из выражений a, b, c, d тоже содержит неизвестные. Второй антецедент вычисляет определитель g системы, преобразуя его нормализатором раскрывания скобок "стандплюс". Равенство g нулю означает зависимость системы; в этом случае заменяющий терм сохраняет лишь первое уравнение. Для срабатывания приема необходимо, чтобы каждое слагаемое выражения g содержало не более одной неизвестной. Уровень срабатывания равен 3.

Попытка разложения на множители разности левых частей уравнений с совпадающими ненулевыми правыми частями

Равенство ненулевых правых частей уравнений может оказаться подсказкой возможности их декомпозиции путем разложения на множители разности левых частей.

$\forall_{abcd}(c = a \ \& \ c - d = b \to d = a \leftrightarrow b = 0)$

Прием применяется в посылках задачи на исследование, не имеющей цели "известно". Правая часть a уравнений $c = a, d = a$ не содержит неизвестных; обе левые части - содержат. Второй антецедент предпринимает попытку разложения разности левых частей на множители с помощью нормализатора "видумножение". Проверяется, что результат b имеет хотя бы два неизвестных сомножителя. Указатели "коммутативно(фикс(1))", "коммутативно(фикс(0 1))" определяют идентификацию частей уравнений без их перестановки. Указатель "первыйключ(простыемножители равно(x3 x1))" играет роль защелки, блокирующей повторные попытки применения приема к той же самой паре уравнений. После идентификации сопровождающего уравнения $c = a$ проверяется отсутствие комментария (простыемножители $c = a$) к текущему уравнению $d = a$. Здесь же этот комментарий вводится. Прием срабатывает на уровне 5 и снабжен сравнительно сильным ограничителем трудоемкости.

Линейная комбинация двух уравнений с целью получения уравнения, линейного по одной неизвестной

Если в каждом из двух уравнений выделяется единственный член, нелинейный относительно некоторой неизвестной, то рассматривается линейная комбинация уравнений, в которой данные нелинейные члены взаимно уничтожаются. Коэффициенты линейной комбинации могут содержать прочие неизвестные.

$\forall_{abcdefgh}(ag + b = c \ \& \ ah + d = e \ \& \ f = bh - dg - ch + eg \to f = 0)$

Прием применяется в задачах на исследование, имеющих более одной неизвестной и не имеющих цели "известно". Первые два антецедента идентифицируются с уравнениями. Указатель "контекст(неизвестная (x9) линейно(x2 x9) входит(x9 x2)входит(x9 x4)линейно(x4 x9)не(входит(x9 x7))не(входит(x9 x8))не(линейно(x1 x9)))" определяет выбор некоторой неизвестной x9, относительно которой линейны выражения b, d и нелинейно выражение a. При этом выражения g, h (коэффициенты линейной комбинации) не содержат x9. Проверяется отсутствие другого уравнения, линейного по x9. Третий антецедент, выделенный указателем "идентификатор", определяет левую часть f линейной комбинации. Эта левая часть обрабатывается нормализатором раскрывания скобок "стандплюс" и нормализатором сумм с неизвестными "уравнплюс". Проверяется, что она содержит x9, линейна относительно x9 и имеет не более двух неизвестных. Прием срабатывает на уровне 8.

Для сохранения информации о взаимной зависимости уравнений прием снабжен указателями "прообраз(1 посылки(не(равно(x8 0))))", "прообраз(2 посылки(не(равно(x7 0))))". Они вводят комментарии (прообраз ...), сообщающие, что одно из исходных уравнений является следствием нового уравнения и другого исходного уравнения. Перед вводом комментария проверяется отличие от нуля соответствующего коэффициента линейной комбинации.

Системы уравнений, встречающиеся в задачах на исследование, имеющих цель "известно"

Предшествующие приемы, связанные с символом "плюс", относились к задачам по элементарной алгебре. Оказалось, что многие из них нецелесообразно применять в других разделах, несмотря на то, что там встречаются те же самые алгебраические операции и отношения. Для обеспечения разумных действий пришлось создавать дубликаты алгебраических приемов, снабжая их совсем другим управлением. Например, так произошло в геометрии, где незаблокированными остались лишь совсем простые алгебраические приемы. Лишь после того, как в геометрической задаче созревает "алгебраическая" система уравнений, предпринимается обращение к решению вспомогательной задачи и начинают работать обычные приемы элементарной алгебры.

Отличительным признаком задач на исследование, требующих особого управления алгебраическими приемами, является цель "известно". Приведем серию отнесенных к символу "плюс" приемов решения уравнений, используемых в таких задачах. Важной характеристикой уравнения здесь становится множество встречающихся в нем неизвестных числовых атомов. Таковыми считаются либо числовые неизвестные, либо атомарные числовые выражения $f(A_1 \dots A_n)$ с неизвестными. В последнем случае хотя бы один из операндов A_i должен принимать нечисловые значения. Примеры числовых атомов - "расстояние($A\ B$)", "масса(A)", и т.п.

1. Рассмотрение линейной комбинации уравнений. Для задач с целью "известно" созданы несколько приемов, использующих линейную комбинацию уравнений.

   $\forall_{abcdefghp}(ag + b = c \ \& \ \neg(g = 0) \ \& \ f = bh - dg - ch + eg \ \& \ p = f \to ah + d = e \leftrightarrow p = 0)$

   Первый антецедент идентифицируется со вторым уравнением. Находятся "подобные члены" ag и ah, у которых переменная a идентифицируется с невырожденным произведением всех неизвестных сомножителей, так что коэффициенты g, h известны. Требуется, чтобы все неизвестные хотя бы одного из уравнений содержались в другом. Третий антецедент формирует левую часть f комбинации уравнений и применяет к ней нормализатор раскрывания скобок "стандплюс". На этом этапе проверяется, что f имеет ровно один неизвестный числовой атом. При положительном результате проверки обрабатывается четвертый антецедент, в котором выполняется попытка разложить f на множители. Проверяется, что результирующее утверждение не имеет заголовка "или". Это происходит из-за того, что преждевременный разбор случаев в задачах с целью "известно" нежелателен, и приводящие к появлению дизъюнкций приемы срабатывают обычно на более высоких уровнях. Проверяется, что все числовые атомы с заголовками "расстояние", "угол" хотя бы одного из уравнений входят в другое уравнение; что прием не приводит к появлению уравнения с модулем, если исходное уравнения модуля не имело и что преобразуемое уравнение имеет более одного числового атома. Последнее выявляет цель приема: свести количество числовых атомов уравнения до одного. Если уравнение имело вид равенства отличного от переменной числового атома выражению, не имеющему таких атомов, то прием блокируется. Уровень срабатывания равен 4.

   Эта же теорема приема порождает еще несколько приемов, отличающихся друг от друга уровнем срабатывания и прочими фильтрами. В них добавлена обработка выражения f упрощенным нормализатором разложения на множители "факторизация". В задачах с целью "известно" понижение уровня срабатывания может привести к отключению приема, так как на малом уровне еще не успеют появиться все необходимые для него данные. Поэтому часто приходится повышать уровень срабатывания даже вполне "естественного" приема, а иногда и брать несколько разных уровней срабатывания.

На уровнях 6 и 10 срабатывает версия приема, в которой оба уравнения не имеют числовых атомов, отличных от переменных, а результирующее уравнение либо имеет меньше числовых неизвестных, чем исходное, либо оно линейно, а исходное нелинейно. Наконец, имеется версия приема, срабатывающая на уровне 11. В ней требуется, чтобы исходное уравнение имело более одного числового атома, а каждый неизвестный сомножитель выражения f имел ровно один неизвестный числовой атом.

Еще один прием линейной комбинации двух уравнений относится к паре однородных уравнений:

$\forall_{abcdep}(\neg(p=0)\ \&\ \neg(d=0)\ \&\ db+ep=0 \rightarrow ab+cp=0 \leftrightarrow cd-ae=0)$

Выводится утверждение о линейной зависимости уравнений, имеющих ненулевое решение. Выражение p содержит неизвестные, причем хотя бы один из коэффициентов c, e тоже содержит неизвестные. Оба коэффициента a, d при b известны, т.е. неизвестные выражения в новом уравнении не перемножаются. Относительно b никаких допущений не делается. Проверяется, что число неизвестных у нового уравнения меньше, чем у исходного и что c, e не содержат неизвестных тригонометрических операций. Уровень срабатывания равен 3.

Последний прием раздела выражает два неизвестных выражения A, B через другое неизвестное выражение r, решая для этого простейшую систему двух линейных уравнений:

$\forall_{ABabcpqrm}(\neg(b=0)\ \&\ m=aq+pbc\ \&\ \neg(m=0) \rightarrow pA+qB=r\ \&\ (aA)/b=cB \leftrightarrow mB=ar\ \&\ mA=bcr)$

Здесь A, B содержат неизвестные; a, b, c, p, q известны; r ожет содержать неизвестные. Уровень срабатывания равен 7.

2. Перенесение в правую часть всех известных слагаемых.

   $\forall_{abc}(a+b=c \leftrightarrow a=c-b)$

   Выражение b идентифицируется с суммой всех известных слагаемых той части уравнения, которая содержит неизвестные. Остаток a невырожден; выражение c известно. Прием срабатывает на уровне 1.

3. Решение системы двух линейных уравнений с двумя неизвестными выражениями.

   Для ускоренного разрешения соотношений, выводимых в задачах с целью "известно", создан ряд приемов, усматривающих стандартного вида подсистемы и сразу выписывающих их ответ. Простейший случай такого рода - линейная подсистема из двух уравнений с двумя неизвестными:

   $\forall_{abcdefxyp}(p=bd-ae\ \&\ \neg(p=0) \rightarrow ax+by=c\ \&\ dx+ey=f \leftrightarrow x=(bf-ce)/p\ \&\ y=(cd-af)/p)$

Выражения x, y содержат неизвестные; a, b, c, d, e, f известны. Правые части выражений для x, y упрощаются с помощью вспомогательных задач на преобразование. Уровень срабатывания приема равен 3.

4. Система из линейного уравнения и уравнения для суммы квадратов.

   Созданы два приема, первый из которых определяет произведение двух неизвестных выражений по их сумме и сумме их квадратов:

   $\forall_{abxy}(x + y = a \to x^2 + y^2 = b \leftrightarrow 2xy = a^2 - b)$

   Выражения a, b известны. Проверяется наличие посылки, содержащей произведение x, y. Уровень срабатывания равен 5.

   Второй прием, срабатывающий на уровне 6, явно определяет x, y:

   $\forall_{abcdxypqr}(p = da^2 + db^2 - c^2 \;\&\; 0 \leq a \;\&\; 0 \leq b \;\&\; q = a^2 + b^2 \;\&\; 0 < q \;\&\; r = \sqrt{p} \to ax + by = c \;\&\; x^2 + y^2 = d \leftrightarrow 0 \leq p \;\&\; (x = (br + ac)/q \;\&\; y = (-ar + bc)/q \;\vee\; x = (-br + ac)/q \;\&\; y = (ar + bc)/q))$

5. Уравнение для суммы двух неизвестных выражений и два уравнения для квадратов этих величин.

   Из двух уравнений для квадратов извлекается уравнение для разности квадратов, которое делится на уравнение для суммы. В итоге определяется разность величин:

   $\forall_{abcdexy}(ax^2 + b = c \;\&\; x + y = e \to ay^2 + b = d \leftrightarrow a(x - y)e = c - d)$

   $\forall_{abcdexy}(x^2 + a = b \;\&\; x + y = e \;\&\; f = b - a - d + c \to y^2 + c = d \leftrightarrow (x - y)e = f)$

   Первый прием срабатывает на уровне 5. Предполагается, что выражения a, c, d, e известны. Уровень срабатывания второго приема равен 6. Его третий антецедент определяет разность f группы членов уравнений с квадратами; после упрощения этой разности проверяется отсутствие в ней неизвестных. Выражение e предполагается известным.

6. Перенесение в левую часть подобных неизвестных слагаемых.

   Для задач на исследование, имеющих цель "известно", заблокирован прием группировки всех неизвестных членов в левой части уравнения, а всех известных - в правой. Уравнения здесь обычно играют роль указателей на наличие связей между числовыми атомами. Разрешение их относительно числовых атомов выполняется лишь в простейших случаях, причем приемами, ориентированными на конкретные типы таких атомов. Большинство обычных приемов решения уравнений отключены. Для решения более сложных подсистем уравнений используется обращение к вспомогательной "алгебраической" задаче. Это и делает необязательной группировку в одной части всех неизвестных членов. Тем не менее, приведение находящихся в различных частях уравнения подобных членов может настолько упростить его, что далее сработает какой-либо из приемов, выполняющих немедленное разрешение относительно неизвестного числового атома. Поэтому были созданы следующие два приема:

   $\forall_{abcdepq}((ab)/p + c = (db)/q + e \leftrightarrow (a/p - d/q)b + c = e)$

   $\forall_{abcdep}(a(pb + c) = db + e \leftrightarrow (ap - d)b + ac = e)$

   Здесь b содержит неизвестные. В первом приеме a,d известны, во втором - a,p известны. Уровень срабатывания приемов равен 1.

В простейшем случае, когда подобные члены имеют численные коэффициенты, отменяется блокировка применения приема к посылкам, используемым для сопровождения по о.д.з. Теорема приема имеет здесь следующий вид:

$\forall_{abcx}(ax + b = cx \leftrightarrow b = (c - a)x)$

a, c - десятичные числа; x содержит неизвестные. Уровень срабатывания приема равен 2.

7. Уравнения для суммы и произведения неизвестных величин.

   Это еще один прием для разрешения стандартной подсистемы:

   $\forall_{abcdxy}(c = a^2 - 4b \;\&\; d = \sqrt{c} \rightarrow x + y = a \;\&\; xy = b \leftrightarrow 0 \leq c \;\&\; (x = (a - d)/2 \;\&\; y = (a + d)/2 \;\vee\; x = (a + d)/2 \;\&\; y = (a - d)/2))$

   Выражения x, y содержат неизвестные. Дискриминант c не зависит от неизвестных внешней задачи на описание; выражения a, b не содержат геометрических числовых атомов "расстояние", "угол", "площадь". Уровень срабатывания приема равен 13.

8. Решение линейного уравнения для выражения числового атома через числовые параметры.

   $\forall_{abcd}(\neg(b = 0) \rightarrow ab + c = d \leftrightarrow a = (d - c)/b)$

   Выражение a представляет собой числовой атом, не являющийся константой либо переменной. Выражения b, c, d не содержат числовых атомов такого вида. Отбрасываются случаи геометрических числовых атомов "расстояние", "угол", "площадь", так как для них имеются свои приемы решения линейных уравнений, обладающие более детализированным управлением (см. главу, посвященную элементарной геометрии). Проверяется, что либо d не является неизвестной, либо левая часть уравнения содержит неизвестные. Проверяется также отсутствие внешних описателей. Прием срабатывает на уровне 4.

9. Группировка в одной части уравнения подобных членов с одинаковым числовым атомом.

   $\forall_{abcdA}(aA + b = cA + d \leftrightarrow (a - c)A + b = d)$

   A - неконстантный числовой атом, отличный от переменной. Коэффициенты a, c не содержат числового атома, отличного от переменной. Так как в геометрии группировка подобных членов с числовыми атомами выполняется другими приемами, данный прием заблокирован для атомов вида "расстояние", "угол", "площадь", "объем". Уровень срабатывания равен 3.

10. Сложение неизвестных дробей в уравнении, не имеющем нечисловых параметров.

    Имеются три приема с одной и той же теоремой:

    $\forall_{abc}(c = a - b \rightarrow a = b \leftrightarrow c = 0)$

    Все числовые атомы уравнения суть переменные. Антецедент обращается к нормализатору "видумножение", обеспечивающему сложение дробей. Первый прием срабатывает на уровне 4. В этом случае проверяется, что в уравнении выделяется сумма с дробным слагаемым, имеющим неизвестный знаменатель, и нет дроби с другим неизвестным знаменателем. Проверяется также, что обе части равенства имеют числовые значения, а число неизвестных уравнения не менее 3. Второй прием срабатывает на уровне 6, третий - на уровне 11. Оба эти приема предполагают наличие суммы с неизвестным дробным слагаемым. Число неизвестных данного уравнения должно быть не меньше 3, а число всех неизвестных задачи, входящих в уравнения без невырожденных (т.е. отличных от переменных) числовых атомов, не менее 4. Второй прием срабатывает только в задачах, использующих некоторую прямоугольную систему координат; третий - не требует этого ограничения.

11. Сложение с константой дроби, имеющей константный знаменатель и константное слагаемое числителя.

    Сумма преобразуется так, чтобы можно было выполнить арифметические действия:

    $\forall_{abcd}((a+b)/c+d=(a+b+cd)/c)$

    Выражения b, c, d идентифицируются с константами. Уровень срабатывания приема равен 2. Арифметические действия (сложение b с результатом умножения c на d) выполняются нормализаторами общей стандартизации "нормплюс", "нормумножение".

12. Сложение трех уравнений для исключения слагаемых.

    Усматривается "циклическое" сокращение группы неизвестных слагаемых при сложении либо вычитании трех уравнений:

    $\forall_{abcdefpqr}(a-b+d=p\ \&\ b-c+e=q\ \&\ c-a+f=r\rightarrow d+e+f=p+q+r)$

    $\forall_{abcdefgp}(a+b=c\ \&\ d-a=e\ \&\ p=f+c-d+e-g\rightarrow f+b=g\leftrightarrow p=0)$

    В первом приеме выражения d, e, f известны; p, q, r - содержат неизвестные. Сокращаемые выражения a, b, c содержат символ "расстояние". Этот прием выводит следствие и срабатывает на уровне 9. Второй прием, срабатывающий на уровне 7, выполняет эквивалентное преобразование одного уравнения с помощью двух других. Выражение c известно; до преобразования уравнение имело числовые атомы, отличные от констант и переменных, а после преобразования - не имеет. Левая часть p результирующего уравнения упрощается вспомогательной задачей на преобразование.

13. Сложение двух уравнений для определения суммы, встречающейся в другом уравнении.

    $\forall_{abmnk}(a+m=n\rightarrow b-m=k\leftrightarrow a+b=n+k)$

    Выражения k, n известны; выражения m, a, b содержат неизвестные. Некоторое уравнение задачи имеет сумму, содержащую слагаемые a, b. Выражения a, b не имеют числовых атомов, отличных от констант и переменных. Прием срсбатывает на уровне 6.

    Следующий прием того же типа явно включает ссылку на третье уравнение:

    $\forall_{xabcdefgm}(x=a+b\ \&\ ma+d=e\ \&\ nb+f=g\ \&\ m=n\ \&\ \neg(m=0)\ \&\ p=d+f\ \&\ q=e+g\rightarrow x=(q-p)/m)$

    Здесь выражение m известно; x - неизвестная; a, b содержат неизвестные. Первые три антецедента идентифицируются с уравнениями; четвертый, шестой и седьмой выделены указателями "идентификатор"; пятый обрабатывается проверочным оператором. Прием находит выражение для x через известные параметры после того, как неизвестные слагаемые второго и третьего уравнений сокращаются. Общий коэффициент при a и b во втором и третьем уравнениях идентифицируется раздельно: сначала как переменная m, затем - как переменная n. Это предотвращает отнесение к данному коэффициенту вообще всех общих множителей данных членов. a и b идентифицируются из первого уравнения, причем указатель "операнд(x1 фикс(1 2))" определяет идентификацию a с отдельным слагаемым. Лишь после этого переменные m, n будут идентифицированы как остаточные множители при a, b, а четвертый антецедент проверит совпадение m с n. Шестой и седьмой антецеденты определяют вспомогательные суммы p, q. После обработки нормализатором "нормплюс" происходит проверка того, что эти суммы не содержат неизвестных. Уровень срабатывания приема равен 6.

14. Решение линейного уравнения относительно известного численного параметра при контроле подслучая.

    Если имел место внешний разбор случаев, то предпринимается разрешение относительно известных числовых параметров x линейных соотношений, не содержащих неизвестных. После подстановки найденного для x выражения в остальные посылки повышается вероятность обнаружения нереализуемого подслучая.

    $\forall_{abcx}(\neg(a = 0) \rightarrow ax + b = c \leftrightarrow x = (c - b)/a)$

    x идентифицируется с переменной - известным числовым параметром; выражения a, b, c не содержат переменной x, неизвестных и неконстантных числовых атомов, отличных от переменных. Чтобы отсечь ситуации, в которых равенство уже дает явное выражение для параметра c, проверяется, что c либо не является переменной, либо встречается в противоположной части равенства. Результирующее равенство для x снабжается комментарием "ориентацияравенства", блокирующим попытки противоположной его ориентации. Прием срабатывает на уровне 2.

15. Усмотрение равенства нулю произведения двух двучленов.

    $\forall_{abcd}(ab + ad + bc + cd = 0 \leftrightarrow (a + c)(b + d) = 0)$

    Имеются две версии этого приема. Первая срабатывает на уровне 5, причем проверяется, что уравнение не имеет числовых атомов, отличных от переменных и констант. Вторая срабатывает на уровне 9 без указанного ограничения.

16. Перенесение в левую часть всех ненулевых членов уравнения, если обе его части линейны относительно некоторой неизвестной.

    $\forall_{abcdef}(a = (bx + c)d/e + f \leftrightarrow a - (bx + c)d/e - f = 0)$

    Переменная x является неизвестной внешней задачи на описание. Она входит в левую часть a и не входит в выражения b, c, d, e. Выражения a, f линейны по x. Уравнение не имеет неконстантных числовых атомов, отличных от переменных. В действительности этот прием применяется крайне редко - только в ситуациях, когда число вхождений неизвестных тригонометрических операций в уравнение не менее 6. Фильтры "не(входит(расстояние корень))", "не(входит(угол корень))" являются ускоряющими: они заведомо выполнены из-за требования отсутствия нетривиальных числовых атомов, однако размещаются в начале программы и сразу отсекают попытки рассмотрения геометрических соотношений. Уровень срабатывания приема равен 4.

17. Внесение минуса под знак суммы.

    Если одна из частей уравнения представляет собой сумму, расположенную под знаком "минус", а другая часть содержит неизвестные, то минус вносится под сумму:

    $\forall_{abc}(-(a+b)=c \leftrightarrow -a-b=c)$

    Выражения a, c содержат неизвестные; существует неизвестный числовой атом, входящий как в a, так и в c. Прием срабатывает на уровне 4.

Сложение либо вычитание целочисленных уравнений для разложения на множители левой части

Приемы усматривают такие два целочисленных уравнения, что при их сложении либо вычитании левая часть оказывается произведением двух целочисленных двучленов, а правая - целым числом. Чтобы последнее было возможно меньшим, при равенстве знаков правых частей уравнения вычитаются, а при различии - складываются. Дальнейшее решение уравнения будет выполняться путем перебора делителей правой части.

$\forall_{abcdpq}(0 \leq pq \ \& \ ac+bd=q \rightarrow ab+cd=p \leftrightarrow (a-d)(b-c)=p-q)$

$\forall_{abcdpq}(pq<0 \ \& \ ac+bd=q \ \& \ ab=cd=p \leftrightarrow (a+d)(b+c)=p+q)$

Уравнения расположены в условиях задачи на описание, имеющей условие вида "целое(x)" либо "натуральное(x)" для некоторой неизвестной x, входящей в преобразуемое уравнение. p, q идентифицируются с целочисленными константами. Приемы срабатывают на уровне 4.

Подбор примера для линейного однородного уравнения с двумя неизвестными

Прием указывает нулевое решение:

$\forall_{abxy}(x=0 \ \& \ y=0 \rightarrow ax+by=0)$

Заголовок приема - "подборзначений". Уравнение $ax+by=0$ является условием задачи на описание, имеющей цель "пример" и не имеющей цели "независит ...". x, y - неизвестные, причем вне данного уравнения они могут встречаться только в утверждениях "число(x)", "число(y)". Антецеденты выделены указателями "подборзначений" - они помещаются в задачу вместо рассматриваемого уравнения. Уровень срабатывания равен 1.

Усмотрение равенства неизвестного квадрата суммы либо квадрата разности известному выражению

Приемы представляют левую часть уравнения в виде произведения квадрата неизвестной суммы на известное выражение, подготавливая таким образом возможность извлечения корня.

$\forall_{abcd}(da^2+2abd+db^2=c \leftrightarrow (a+b)^2d=c)$

$\forall_{abcd}(da^2 - 2abd + db^2 = c \leftrightarrow (a-b)^2 d = c)$

Выражения c, d известны; уравнение расположено в списке условий задачи на описание. Приемы срабатывают на уровне 1.

Активизация анализатора "сложениеуравнений"

Пакетный анализатор "сложениеуравнений" создан для вывода линейных комбинаций уравнений в задаче на исследование. Часто он и выполняет основную часть работы по решению системы уравнений, обеспечивая поиск удачной их комбинации. Активизация анализатора реализуется двумя приемами с заголовком "замечание", имеющими следующую теорему:

$\forall_{abc}(a + b = c \rightarrow \emptyset)$

Первый прием срабатывает на уровне 3 в задаче на исследование, не имеющей цели "известно". Выражения a, b содержат неизвестные; общее число неизвестных задачи должно быть более одной. Не должно существовать линейного по всем своим неизвестным уравнения. Указатель "анализатор(сложениеуравнений уровень(1)лимит(вариант(и(равно(числонеизвестных 2)меньше(80 число(посылка(x4)заголовок(x4 равно)входит(x5 x4))))6000000 3000000)))" реализует обращение к анализатору. Если число неизвестных задачи равно 2, а суммарная длина ее уравнений больше 80, то ограничитель трудоемкости равен 6000000 шагов работы интерпретатора ЛОСа, иначе он равен 3000000.

Второй прием аналогичен первому и срабатывает на уровне 7. Требование на нелинейность уравнений здесь отсутствует, ограничитель трудоемкости равен 6000000.

Разложение на множители коэффициента, имеющего вид суммы

Прием выполняет разложение на множители известных сумм - оснований степени в коэффициентах членов уравнений. Он применяется в задачах, имеющих более одной неизвестной, и в ряде случаев позволяет сократить уравнение на известный множитель. Теорема приема имеет следующий вид:

$\forall_{abcdefg}(g = a + b \rightarrow (a+b)^c d + e = f \leftrightarrow g^c d + e = f)$

Уравнение либо является условием задачи на описание, либо посылкой задачи на исследование. Сумма $a + b$ и выражение f не содержат неизвестных; выражения d, e - содержат. Антецедент использует для разложения на множители нормализатор "видумножение". Показатель степени c - натуральная константа, возможно, равная 1. Проверяется, что результат g разложения на множители имеет, с точностью до знака, мультипликативный заголовок ("умножение" либо "степень" либо "дробь"). Уровень срабатывания приема равен 2.

Вынесение за скобку общего известного множителя всех неизвестных слагаемых левой части уравнения

Теорема приема имеет вид $\forall_{abc}(a(b+c) = ab+ac)$. Заголовок приема - "первыйтерм". Наличие указателя "набор(втаройтерм)" дает возможность одновременно обрабатывать все слагаемые. Прием применяется к сумме, являющейся одной из частей уравнения задачи на описание либо на исследование. При наличии цели "известно" противоположная часть уравнения должна не содержать неизвестных. Задача имеет более одной неизвестной. Указатель "перечень(x1 известно(x1))" определяет идентификацию переменной a с произведением всех известных множителей общего делителя слагаемых.

Создан еще один прием с той же теоремой, применяемый к сумме, являющейся одной из частей посылки задачи на доказательство. Противоположная часть суммы - либо известная (в случае задачи на доказательство это означает, что все ее переменные зарегистрированы в комментарии "известно"), либо имеет своим множителем выражение a. Как и выше, используются указатели "набор(второйтерм)" и "перечень(x1 известно(x1))".

Приемы срабатывают на уровне 1. Обычно после этого происходит деление обеих частей уравнения на выделенный известный множитель.

Вынесение за скобку общего неизвестного множителя всех слагаемых левой части уравнения при нулевой правой части

Если правая часть уравнения нулевая, то имеет смысл выделять не только общие известные, но и общие неизвестные множители слагаемых левой части. Теорема приема - та же, что выше: $\forall_{abc}(a(b+c)=ab+ac)$. Сумма является одной из частей уравнения задачи на описание, имеющего нулевую противоположную часть. Общий множитель a содержит неизвестные. Прием имеет указатель "набор(второйтерм)" и срабатывает на уровне 1.

По этой же теореме создан аналогичный прием, срабатывающий в задачах на исследование и на доказательство. Не требуется, чтобы a содержало неизвестные, однако сумма должна быть неконстантной. Уровень срабатывания равен 2.

Условия равенства нулю суммы и разности

Происходит простейшая стандартизация двух уравнений задачи на описание:

$\forall_{ab}(a+b=0 \ \& \ a-b=0 \leftrightarrow a=0 \ \& \ b=0)$

Указатель "нормзнака(x2 минус плюс)" позволяет идентифицировать выражение b как сумму нескольких слагаемых; при этом $-b$ идентифицируется в виде суммы этих слагаемых с измененными знаками. Уровень срабатывания равен 2.

Линейная комбинация двух уравнений с одной неизвестной

Если задача имеет два уравнения относительно одной и той же неизвестной, то иногда можно упростить решение, рассмотрев такую линейную комбинацию уравнений, которая уничтожает часть неизвестных слагаемых. Этот способ решения может оказаться единственно возможным. Теорема приема имеет следующий вид:

$\forall_{abcdpqrA}(\neg(a=0) \ \& \ ab+c=d \ \& \ (aq-cp=ar-dp)=A \rightarrow pb+q=r \leftrightarrow A)$

Текущая задача на описание имеет единственную неизвестную. Второй антецедент идентифицируется с дополнительным уравнением. Выражение b, исключаемое с помощью линейной комбинации, содержит неизвестные. Выражения a, p, d, r (коэффициенты линейной комбинации и правые части уравнений) известны. Третий антецедент, выделенный указателем "идентификатор", определяет заменяющее утверждение A. Его левая часть обрабатывается вспомогательной задачей на описание, имеющей ту же неизвестную, что и текущая задача. Максимальный уровень вспомогательной задачи равен 5. Прием имеет средний ограничитель трудоемкости. Фактически, он реализует попытку "сразу" найти ответ, подставляя в случае успеха явно разрешенное относительно неизвестной утверждение A.

Линейная комбинация двух уравнений для исключения параметра

Если в задаче с одной неизвестной возникли два уравнения, имеющие известный параметр, причем существует их линейная комбинация, устраняющая этот параметр, то целесообразно сразу перейти к ее рассмотрению. Это позволяет исключить параметр из выражения для неизвестной и может сильно упростить дальнейшие выкладки. Теорема приема реализует данную схему действий для одного частного случая:

$\forall_{ABbcd}(\neg(b = 0) \ \& \ A = bc \rightarrow B = cd \leftrightarrow Ad - Bb = 0)$

Решается задача на описание с единственной неизвестной. Второй антецедент идентифицируется с дополнительным уравнением. Левые части A, B содержат неизвестные; правые части известны. Выражения b, d константные. Существует известный параметр, встречающийся в выражении c и не встречающийся в A, B. Выражение B содержит хотя бы одну из операций "плюс", "умножение", "степень", "минус".

Уменьшение числа слагаемых с помощью тождества из контекста

При доказательстве числового равенства может использоваться его линейная комбинация с равенством из посылок, приводящая к уменьшению числа слагаемых. Создан простейший прием такого типа:

$\forall_{abcd}(d = a - c \ \& \ b + c = 0 \rightarrow a + b = 0 \leftrightarrow d = 0)$

Преобразуемое равенство является условием задачи на доказательство и не содержит неизвестных. Второй антецедент идентифицируется с равенством из посылок; первый - вычисляет разность d левых частей. Проверяется, что число ее слагаемых меньше числа слагаемых исходного равенства. Для ускорения отсечения проверяется, что сокращаемая часть b имеет более одного слагаемого. Прием срабатывает на уровне 3.

Попытка разложения на множители для упрощения выражения

При разложении на множители сумма может приобрести более компактный вид. Попытка такого ее упрощения применяется в условиях задач на преобразование. Однако, имеется множество ситуаций, в которых преобразование суммы в произведение либо нецелесообразно, либо реализуется другим приемом. Поэтому описываемый здесь прием имеет большое число фильтров, возникавших в процессе обучения постепенно. Теорема приема такова:

$\forall_{abc}(c = a + b \rightarrow a + b = c)$

Сумма расположена в условии задачи на преобразование, имеющей цель "упростить", причем запрещается наличие любой из следующих целей:

1. "раскрытьскобки" (цель предполагает обратное действие)
2. "разложитьнамножители" (действие выполняется другим приемом)
3. "нормИнтеграл" (вспомогательная задача на преобразование подынтегрального выражения; в этом случае предпочтительнее работать с суммой)
4. "вычислить" (задача на программирование требует особого управления преобразованиями)

5. "частнпроизв" (вспомогательная задача на упрощение результата кратного дифференцирования - успешная попытка разложения на множители маловероятна)

6. "редуцирование" (вспомогательная задача логического вывода теорем, требующая особого управления преобразованиями)

Антецедент обращается для разложения на множители к нормализатору "видумножение". Имеется средних масштабов ограничитель трудоемкости, несколько увеличенный для суммы не более чем трех слагаемых. Дополнительно накладываются следующие ограничения:

1. Либо результат c короче исходной суммы, либо решается задача на упрощение известного подвыражения ответа задачи на описание, рассматриваемая сумма служит сомножителем некоторого произведения, и после подстановки в это произведение выражения c удается добиться дальнейшего его упрощения (например, за счет перемножения степеней с одинаковым основанием).

2. Сумма не имеет дробных слагаемых (в этой ситуации работает другой прием).

3. Сумма не является операндом степени (для разложения на множители основания степени имеются другие приемы; показатель степени обычно приводится к виду суммы).

4. Сумма не является сомножителем числителя либо знаменателя дроби (в этой ситуации работают другие приемы).

5. Сумма не расположена под описателями "класс", "отображение" (в этих случаях требуется особое управление преобразованиями, чтобы не нарушить стандартного вида уравнений, определяющих множества либо функции).

6. Сумма не является операндом тригонометрической операции (здесь принята стандартизация, преобразующая выражения к виду сумм).

Указатель "попытка(видумножение копия(плюс(x1 x2)))" определяет проверку отсутствия комментария (видумножение $a+b$), указывающего на ранее предпринимавшуюся неудачную попытку разложения на множители с упрощением выражения. Служебное слово "копия" означает, что в комментарии будет использована не обработанная нормализатором сумма, а ее исходная версия. Если прием не срабатывает, то при откате указанный комментарий вводится. Уровень срабатывания равен 4.

Вынесение за скобку общего множителя всех слагаемых

Прием преобразует сумму, расположенную под описателем "отображение". Теорема имеет вид $\forall_{abc}(ab + ac = a(b + c))$. Сумма находится в условии задачи на преобразование, имеющей цель "упростить". Она не является операндом степени либо тригонометрической операции. Уровень срабатывания равен 6.

Обращение к оператору "видумножение" для разложения на множители

Если сумма расположена в условии задачи на преобразование, имеющей цель "разложитьнамножители", причем путь к ней от корня условия проходит лишь через мультипликативные операции, то предпринимается обращение к нормализатору "видумножение":

$\forall_{abc}(c = a + b \to a + b = c)$

Проверяется отсутствие цели "видУмножение", уточняющей, что множители могут быть комплексными. Проверяется, что результат c обращения к нормализатору либо имеет мультипликативный заголовок ("умножение","дробь","степень", быть может после отбрасывания минуса), либо не имеет вида суммы и короче исходного выражения. Обращение к нормализатору "видумножение" сопровождается комментариями. Если решается не вспомогательная задача, а взятая непосредственно из задачника, то вводится комментарий "максимум", включающий усиленные средства. Если все входящие в сумму степени имеют натуральные показатели, то вводится комментарий "выделениестепени", блокирующий попытки перехода к степеням с дробными показателями. Уровень срабатывания приема равен 1.

Попытка разложения на множители суммы в левой части равенства нулю

Приемы подраздела относятся только к равенствам, не являющимся уравнениями. Случай уравнений охватывается другими приемами, размещенными в разделе символа "умножение".

Для преобразования отрицания равенства суммы нулю служит следующий прием:

$\forall_{ab}(a = b \to \neg(a = 0) \leftrightarrow \neg(b = 0))$

Преобразуемое отрицание является условием задачи на описание, не используемым для сопровождения по о.д.з. Выражение a имеет заголовок "плюс". Антецедент обращается к нормализатору "видумножение"; проверяется, что результат b имеет мультипликативный заголовок (с точностью до отбрасывания знака). Указатель "попытка(видумножение копия(x1))" блокирует повторные неудачные попытки разложения на множители. Уровень срабатывания равен 4. В результате применения приема исходное отрицание равенства распадется на несколько более простых утверждений.

Следующий прием относится к равенствам без неизвестных, возникающим в редактируемом ответе задачи на описание:

$\forall_{ab}(b = a \to a = 0 \leftrightarrow b = 0)$

Задача имеет цель "редакция"; текущий ее терм не содержит неизвестных. Разложение на множители обеспечивается ослабленным нормализатором "факторизация". Прием применяется даже в тех случаях, когда текущий терм используется для сопровождения по о.д.з. Уровень срабатывания равен 3.

Наконец, имеется прием, декомпозирующий равенство нулю в консеквенте кванторной импликации:

$\forall_{Apqr}(p(x) + q(x) = r(x) \to \forall_x(A(x) \to p(x) + q(x) = 0) \leftrightarrow \forall_x(A(x) \to r(x) = 0))$

Антецедент использует обращение к нормализатору "видумножение". Проверяется, что результат имеет мультипликативный заголовок. После применения приема и стандартизации кванторной импликации происходит упрощению консеквента. Уровень срабатывания равен 3.

Усмотрение ложности числового равенства

В подразделе собраны несколько простых приемов, усматривающих ложность числового равенства путем сравнения его с утверждениями, имеющимися в контексте.

Первый прием использует посылку, выражающую равенство нулю подсуммы рассматриваемой суммы:

$\forall_{ab}(a = 0 \;\&\; \neg(b = 0) \to \neg(a + b = 0))$

Прием имеет заголовок "второйтерм". Первый антецедент непосредственно содержится в контексте; второй обрабатывается проверочным оператором. Попытка применения приема выполняется только для термов задачи, имеющих заголовок "существует". Уровень срабатывания равен 1; введен сильный ограничитель трудоемкости.

Другие два приема относятся к случаям перегруппировки членов равенства:

$\forall_{ab}(\neg(a = b) \to \neg(a - b = 0))$

$\forall_{ab}(\neg(a - b = 0) \to \neg(a = b))$

Первый из них имеет уровень срабатывания 3, второй - 1. Во втором случае оба выражения a, b отличны от 0. Антецеденты берутся непосредственно из контекста.

Попытка разложения на множители для доказательства отрицания равенства

$\forall_{ab}(b = a \;\&\; \neg(b = 0) \to \neg(a = 0))$

Прием имеет заголовок "второйтерм". Утверждение $\neg(a = 0)$ является условием задачи на доказательство. Выражение a представляет собой сумму. Первый антецедент обрабатывает ее нормализатором "видумножение"; проверяется, что результат не является суммой. После этого второй антецедент, выделенный указателем "следствие", обращается к вспомогательной задаче на доказательство утверждения $\neg(b = 0)$. Прием срабатывает на уровне 5.

Нормализатор общей стандартизации "нормплюс"

Прежде всего, нормализатор содержит уже рассмотренные выше простейшие приемы стандартизации сумм: сложение с нулем, сложение десятичных чисел и простых дробей, приведение подобных членов с числовыми коэффициентами. К ним добавлены следующие приемы:

1. Вычисление малой отрицательной степени десятичного числа - коэффициента слагаемого.
$$\forall_{abcd}\left(ab^{-c} + d = \frac{a}{b^c} + d\right)$$
b - десятичное число; c - натуральное, не превосходящее 4. Уровень срабатывания равен 3.

2. Вынесение за скобку общего минуса всех слагаемых. Прием применяется на уровне 1. Если его нужно заблокировать, то при обращении к нормализатору вводится комментарий "извлечение". Других ограничений нет.

3. Внесение минуса под знак суммы, если внешняя операция - сложение. $\forall_{abc}(c - (a + b) = c - a - b)$.

4. Усмотрение равенства суммы нулю из равенства нулю суммы слагаемых с обратными знаками. $\forall_{ab}(a - b = 0 \rightarrow b - a = 0)$. Антецедент берется из списка посылок. Указатели "нормзнака(x1 минус плюс)" и "отрицание(минус x2)" определяют одновременное изменение знаков у всех слагаемых суммы. Прием блокируется комментарием "существпосылки".

5. Приведение подобных членов с дробными радикалами.

   В нормализатор введены несколько приемов, обеспечивающих заведомое упрощение сумм с дробными радикалами:

$$\forall_{abcd}\left(c\sqrt{\frac{a}{b}} + d\sqrt{\frac{b}{a}} = \frac{(ac+bd)\sqrt{\frac{a}{b}}}{a}\right)$$

$$\forall_{abcde}\left(\frac{a\sqrt{b}}{c} + \frac{d}{e\sqrt{b}} = \frac{(abe+cd)\sqrt{b}}{bce}\right)$$

$$\forall_{abcde}\left(\frac{a\sqrt{b}}{c} + \frac{d}{eb^{3/2}} = \frac{(ab^2e+cd)\sqrt{b}}{b^2ce}\right)$$

$$\forall_{abcde}\left(\frac{a}{b\sqrt{c}} + \frac{d}{ec^{3/2}} = \frac{ace+bd}{bec^{3/2}}\right)$$

   Первый прием срабатывает на уровне 3. Предполагается, что выражения c, d не содержат символа "O". В остальных приемах все переменные идентифицируются с десятичными константами. Их уровни срабатывания равны 1.

6. Сложение логарифмов.

   Выполняется сложение либо вычитание логарифмов от десятичных констант по общему основанию:

   $\forall_{abcd}(d\log_a b + d\log_a c = d\log_a(bc))$

   $\forall_{abcd}(-d\log_a b - d\log_a c = -\log_a(bc))$

   $\forall_{abcde}(b = ce \rightarrow d\log_a b - d\log_a c = d\log_a e)$

   $\forall_{abcde}(b = ce \rightarrow d\log_a c - d\log_a b = -d\log_a e)$

   В первых двух приемах b, c - десятичные числа; в двух последующих - натуральные константы. Основание a не содержит неизвестных текущей задачи. Антецеденты последних двух приемов выделены указателем "программа" - они выполняют целочисленное деление b на c. Предпоследний прием срабатывает на уровне 2, остальные - на уровне 1.

7. Сложение простых дробей, выраженных через отрицательные степени.

$$\forall_{abcdef}\left(e = \frac{a}{b^c} + d \rightarrow ab^{-c} + d + f = e + f\right)$$

a, d - целочисленные константы; b, c - натуральные константы, причем показатель степени c меньше 5. Антецедент, выделенный указателем "идентификатор", выполняет арифметические действия и получает простую дробь e. Уровень срабатывания равен 2.

8. Сумма квадратов синуса и косинуса.

   $\forall_{abc}(a(\sin b)^2 + a(\cos b)^2 + c = a + c)$

   Прием срабатывает на уровне 3. Указатели "единица" допускают вырожденные значения переменных a, c; указатель "заменазнака" - изменение знака перед слагаемыми.

9. Сложение дробей с одинаковыми знаменателями.

   $\forall_{abc}(a/b + c/b = (a + c)/b)$

   $\forall_{abcd}((a - b)/c + (d + b)/c = a/c + d/c)$

   В действительности сложение дробей с одинаковыми знаменателями далеко не всегда оказывается целесообразным. Поэтому первый прием срабатывает только при наличии комментария "простейшиедроби". Второй прием дает очевидные упрощения и применяется без ограничений. Уровни срабатывания в обоих случаях равны 1.

10. Сложение двух "O". Прием применяется при вычислении пределов:

    $\forall_a(O(a) + O(a) = O(a))$

11. Символы бесконечности. В различных контекстах могут встречаться суммы с символами плюс-минус бесконечности либо символами мощности бесконечных множеств. Для них предусмотрены следующие приемы:

    $\infty + \infty = \infty$

    $\forall_a(a - \text{число} \to a + \infty = \infty)$

    $\forall_a(a - \text{число} \to a - \infty = -\infty)$

    $-\infty - \infty = -\infty$

    $\forall_a(a - \text{число} \to a + \text{счетное} = \text{счетное})$

    $\forall_a(a - \text{число} \to a + \text{континуум} = \text{континуум})$

    $\text{счетное} + \text{счетное} = \text{счетное}$

    $\text{континуум} + \text{континуум} = \text{континуум}$

    $\text{счетное} + \text{континуум} = \text{континуум}$

    Строго говоря, использование нечисловых операндов под суммой некорректно - оно должно восприниматься как фрагмент условной сокращенной записи какого-то другого, логически корректного утверждения. Такая запись оказывается удобной для вычислительных целей и используется лишь в очень локальных контекстах, редко выходя за рамки нормализаторов.

12. Приведение подобных простейших дробей. Этот прием используется нормализатором приведения к виду суммы простейщих дробей. Он выполняет сложение подобных дробных выражений с коэффициентами, не зависящими от переменной, относительно которой берутся простейшие дроби. Для срабатывания

приема необходимо наличие комментария "простейшиедроби". Уровень срабатывания равен 3.

13. Раскрывание скобок по дистрибутивности. Как и в предыдущем случае, прием используется только специальным внешним нормализатором. Им является нормализатор раскрывания скобок "стандплюс". Теорема приема имеет вид $\forall_{abcd}(d + (a + b)c = d + ac + bc)$.

Нормализатор сокращенной записи "упрощплюс"

Нормализатор используется при редактировании ответа задачи. Преобразуемое им выражение имеет заголовок "плюс", причем обрабатывается только корневое вхождение символа "плюс". Теорема приема в таких случаях может идентифицировать лишь часть слагаемых. Помимо приемов устранения вложенных сумм и лексикографического упорядочения операндов, нормализатор выполняет ряд действий, ориентированных на получение более короткой записи:

1. Сложение дробей с равными знаменателями.

   $\forall_{abc}(\neg(c = 0)\ \&\ c - \text{число} \to a/c + b/c = (a + b)/c)$

   Если одно из выражение a, b имеет своим сомножителем натуральную степень неконстантного выражения p, причем некоторое другое слагаемое преобразуемой суммы также имеет свом сомножителем натуральную степень выражения p, то прием блокируется. Это сохраняет возможность последующей группировки с вынесением p за скобку. Уровень срабатывания равен 1.

2. Вынесение за скобку общего множителя слагаемых.

   $\forall_{abc}(ab + ac = a(b + c))$

   Указатель "набор(второйтерм)" определяет выделение общего множителя a из всех слагаемых. Либо сумма неконстантная, либо a есть целочисленная константа. Преобразуемое выражение не должно содержать единиц измерения. Уровень срабатывания равен 1.

   $\forall_{abcdpqr}(a = pr\ \&\ c = qr\ \&\ ab + cd = r(pb + qd))$

   Сумма константная и имеет ровно два слагаемых. Натуральные коэффициенты a и c этих слагаемых имеют невырожденный общий делитель r. Как и в предыдущем случае, не допускаются единицы измерения. Уровень срабатывания равен 2.

3. Вынесение за скобку общего множителя числителей дробей.

   $\forall_{abcde}(\neg(b = 0)\ \&\ \neg(d = 0)\ \&\ b - \text{число}\ \&\ d - \text{число} \to ae/b + ce/d = e(a/b + c/d))$

   Каждое слагаемое суммы - дробь, числитель которой имеет множитель e. При этом неконстантное выражение e определяется как список общих множителей числителей выделенных двух слагаемых. Указатель "наборoperандов(фикс(0 1 1 1)фикс(0 1 2 1))" отменяет использование трудоемкого оператора "алгебрпересечение" при нахождении e. Прием срабатывает на уровне 1.

4. Вынесение за скобку общего множителя знаменателей дробей.

   $\forall_{abcde}(\neg(b = 0)\ \&\ \neg(c = 0)\ \&\ \neg(e = 0)\ \&\ b - \text{число}\ \&\ c - \text{число}\ \&\ e - \text{число} \to a/bc + d/ce = (a/b + d/e)/c)$

Выражение c неконстантное. Прием блокируется, если произошло подразбиение степенного выражения на два неконстантных множителя, один из которых остался в b либо e, а другой вошел в c. Уровень срабатывания равен 1.

5. Вынесение за скобку общего минуса всех слагаемых.

   $\forall_{ab}(-a - b = -(a + b))$

   Прием срабатывает без ограничений на уровне 1.

6. Преобразование суммы логарифмов в логарифм произведения.

   $\forall_{abcd}(0 \leq a \ \& \ 0 \leq b \to d\log_c a + d\log_c b = d\log_c(ab))$

   Уровень срабатывания равен 3.

7. Преобразование разности логарифмов в логарифм дроби.

   $\forall_{abcd}(0 \leq a \ \& \ 0 \leq b \to d\log_c a - d\log_c b = d\log_c(a/b))$

   Уровень срабатывания равен 3.

Нормализатор раскрывания скобок "стандплюс"

Нормализатор обеспечивает преобразование выражения к виду суммы одночленов ("раскрывание скобок"). При наличии комментария "видумножение" он дополнительно преобразует произведения тригонометрических операций в суммы. Это сравнительно большой нормализатор, используемый во множестве приемов.

Формат нормализатора, определяемый справочником "быстрпреобр", не содержит символа "корень", т.е. описываемые ниже преобразования относятся к любым подтермам преобразуемого терма.

1. Устранение вложенных сумм и произведений происходит на уровне 1.

2. Лексикографическое упорядочение слагаемых и сомножителей бывает необходимо для усмотрения одинаковых подвыражений. Однако, его нецелесообразно применять к промежуточным выражениям, быстро изменяемым другими приемами. Поэтому уровни данных приемов сравнительно высоки: сомножители упорядочиваются на уровне 3, слагаемые - на уровне 4.

3. Вычисления с константами. В нормализаторе имеются приемы сложения и умножения десятичных чисел, сложения и сокращения числовых дробей, а также возведения числа в натуральную степень, аналогичные рассмотренным выше. При возведении в степень накладываются следующие ограничения: если основание степени равно 2, то допускаются показатели не большие 26; если основание степени равно 3, то - показатели не большие 12; в прочих случаях - показатели, не большие 8. Кроме того, добавлен прием выделения числового множителя из дробной степени:

   $$\forall_{abcde}\left(0 < b - c \ \& \ b = cd + e \ \& \ f = a^d \to a^{\frac{b}{c}} = fa^{\frac{e}{c}}\right)$$

   Здесь a идентифицируется с десятичной константой; b, c - с натуральными константами. Первые два антецедента выделены указателями "программа". Проверяется, что числитель b больше знаменателя c и выполняется деление с остатком одного на другой. Третий антецедент, выделенный указателем "идентификатор", выполняет возведение числа a в степень d, если она не превосходит 4.

Для этого используется вспомогательный оператор "натурстепень", реализованный на ЛОСе.

4. Простейшие упрощающие тождества. Так как при раскрывании скобок часто возникают разнообразные возможности для очевидного упрощения выражений, в нормализатор пришлось ввести множество приемов такого упрощения. Перечислим эти приемы, ограничиваясь в большинстве случаев лишь названием выполняемого действия:

   (a) Отбрасывание минуса перед нулем.

   (b) Умножение на ноль и на единицу.

   (c) Вынесение минуса за знак умножения.

   (d) Отбрасывание двойного минуса.

   (e) Сложение с нулем.

   (f) Отбрасывание единичного знаменателя.

   (g) Замена на ноль дроби с нулевым числителем.

   (h) Перенесение в числитель дроби множителя перед дробью.

   (i) Деление на дробь.

$$\forall_{abc}\left(\neg(a=0)\ \&\ \neg(c=0)\ \&\ a - \text{число}\ \&\ c - \text{число} \rightarrow \frac{b}{\frac{c}{a}} = \frac{ab}{c}\right)$$

   (j) Деление дроби.

$$\forall_{abc}\left(\neg(a=0)\ \&\ \neg(c=0)\ \&\ a - \text{число}\ \&\ c - \text{число} \rightarrow \frac{b}{ac} = \frac{\frac{b}{c}}{a}\right)$$

   Прием имеет заголовок "замена(первыйтерм стандплюс)".

   (k) Вынесение минуса из знаменателя либо числителя дроби.

   (l) Сокращение дробного выражения.

$$\forall_{abc}\left(\neg(c=0)\ \&\ c - \text{число}\ \&\ \neg(b=0)\ \&\ b - \text{число} \rightarrow \frac{ac}{bc} = \frac{a}{b}\right)$$

   Заметим, что прием не просто отбрасывает явно выделенные общие множители c числителя и знаменателя, но определяет "наибольший общий делитель" числителя и знаменателя, используя процедуру "алгебрпересечение". Эта же процедура находит и остаточные части a, b. Таким образом, будут сокращаться и "замаскированные" общие множители, например, степени с одинаковыми основаниями.

   (m) Замена произведения модулей на модуль произведения.

   (n) Замена частного модулей на модуль дроби.

$$\forall_{abcd}\left(\neg(a=0)\ \&\ \neg(c=0)\ \&\ a - \text{число}\ \&\ c - \text{число} \rightarrow \frac{d|b|}{a|c|} = \frac{d\left|\frac{b}{c}\right|}{a}\right)$$

  (o) Устранение единицы в показателе степени.

  (p) Четная степень модуля.
  $\forall_{ab}(a - \mathrm{rational}\ \&\ \text{числитель}(a) - \mathrm{even} \rightarrow |b|^a = b^a)$

  (q) Произведение нечетной степени модуля на сигнум.
  $\forall_{an}(|a|^n \mathrm{sg} a = a^n)$
  Здесь n идентифицируется с нечетной константой.

  (r) Стандартизация тригонометрических многочленов с применением тождества для суммы квадратов синуса и косинуса, а также тождества для суммы четвертых степеней синуса и косинуса. Так как эти приемы сравнительно громоздки и являются копиями приемов общей стандартизации тригонометрических выражений, рассматриваемых ниже в разделе "Тригонометрия", здесь мы их рассматривать не будем. Громоздкость теорем приемов является следствием такого обобщения указанных выше несложных тригонометрических тождеств, при котором становится возможно применение их в различных "замаскированных" ситуациях.

5. Стандартизирующие тождества. Кроме приведенных выше простейших упрощающих тождеств, нормализатор имеет ряд дополнительных стандартизирующих тождеств, используемых в процессе раскрывания скобок:

  (a) Приведение подобных членов, имеющих числовые коэффициенты. Приемы аналогичны рассмотренным выше приемам общей стандартизации; отдельно рассматриваются случаи сложения подобных дробей, а также подобных дроби и недробного выражений.

  (b) Сокращение числовых множителей в числителе и знаменателе дроби.

$$\forall_{abcd}\left(\neg(c = 0)\ \&\ c - \text{число} \rightarrow \frac{ad}{bc} = \frac{d\frac{a}{b}}{c}\right)$$

  Переменные a, b идентифицируются с десятичными константами. Выражение $\frac{a}{b}$ обрабатывается нормализатором "сокращдроби". Указатель "различны(сокращдроби дробь(x1 x2))" обеспечивает проверку фактического изменения этого выражения. Так как заменяющий терм обрабатывается нормализаторами "нормумножение" и "нормдробь", то в случае дробного результата сокращения будет обеспечено исключение двухэтажной дроби.

  (c) Перемножение числовых оснований степени, имеющих равные показатели.
  $\forall_{abc}(0 \leq a\ \&\ 0 \leq b \rightarrow a^c b^c = (ab)^c)$
  $\forall_{abc}(c - \mathrm{rational}\ \&\ \neg(\text{знаменатель}(c) - \mathrm{even}) \rightarrow a^c b^c = (ab)^c)$
  Переменные a, b идентифицируются с десятичными константами, так что в основании результирующей степени снова оказывается десятичная константа.

  (d) Преобразование степени произведения в произведение степеней. Такой переход рассматривается как общая стандартизация выражения; обратный переход характерен лишь для особых случаев (например, при завершающем редактировании).
  $\forall_{abc}(0 \leq a\ \&\ 0 \leq b \rightarrow a^c b^c = (ab)^c)$
  $\forall_{abc}(c - \mathrm{rational}\ \&\ \neg(\text{знаменатель}(c) - \mathrm{even}) \rightarrow a^c b^c = (ab)^c)$
  В обоих случаях замена выполняется справа налево.

(e) Преобразование степени дроби в частное степеней.

$$\forall_{abc}\left(c - \text{rational} \ \& \ \neg(\text{знаменатель}(c) - \text{even}) \rightarrow \left(\frac{a}{b}\right)^c = \frac{a^c}{b^c}\right)$$

(f) Умножение степеней с одинаковым основанием.

$\forall_{abc}(0 \leq a \rightarrow a^b a^c = a^{b+c})$

$\forall_{abc}(b\text{—rational} \ \& \ c\text{—rational} \ \& \ \neg(\text{знаменатель}(b)\text{—even}) \ \& \ \neg(\text{знаменатель}(c)\text{—even}) \rightarrow a^b a^c = a^{b+c})$

Приемы не применяются, если умножается десятичное число на свою дробную степень.

(g) Деление степеней с одинаковым основанием.

$\forall_{abcde}(b\text{—rational} \ \& \ c\text{—rational} \ \& \ \neg(\text{знаменатель}(b)\text{—even}) \ \& \ \neg(\text{знаменатель}(c)\text{—even}) \ \& \ \neg(e = 0) \ \& \ \neg(a = 0) \ \& \ a - \text{число} \ \& \ e - \text{число} \rightarrow (da^b)/(ea^c) = (da^{b-c})/e)$

$\forall_{abcde}(0 < a \ \& \ a - \text{число} \rightarrow (da^b)/(ea^c) = (da^{b-c})/e)$

Приемы не применяются, если десятичное число делится на свою дробную степень.

(h) Свертка произведения сигнума на число в модуль числа.

$\forall_{an}(a^n \text{sg} a = a^{n-1}|a|)$

n - натуральная константа. Прием применяется лишь в особых случаях: требуется наличие комментария "уравндифф", вводимого при решении дифференциальных уравнений, причем преобразуемый терм должен уже содержать $|a|$.

(i) Повторное возведение в степень.

$\forall_{abc}(0 \leq a \rightarrow (a^b)^c = a^{bc})$

$\forall_{abc}(b\text{—rational} \ \& \ c\text{—rational} \ \& \ \neg(\text{знаменатель}(b)\text{—even}) \ \& \ \neg(\text{знаменатель}(c)\text{—even}) \rightarrow (a^b)^c = a^{bc})$

(j) Выделение множителя из дробной степени суммы.

Прием позволяет выводить из-под дробной степени суммы, для которых можно продолжить процесс "раскрывания скобок".

$\forall_{abcde}(0 < b - c \ \& \ b = cd + e \ \& \ 0 \leq a \rightarrow a^{b/c} = a^d a^{e/c})$

Здесь b, c - натуральные константы, причем c четно; a - выражение с заголовком "плюс". Первые два антецедента выделены указателем "программа"; второй из них выполняет деление с остатком b на c. Проверяется, что неполное частное d не превосходит 3.

(k) Представление произведения неизвестных степеней с численными основаниями в виде степени произведения этих оснований, домноженной на известный коэффициент. Прием создает повторное вхождение степени с неизвестным показателем и численным основанием.

$\forall_{abcdef}(f = ad \rightarrow a^{b+c} d^{b+e} = a^c d^e f^b)$

Основания степени a, d идентифицируются с натуральными константами. f - их произведение, вычисляемое первым антецедентом. Преобразуемое выражение уже содержит степень с основанием f и показателем вида $b+k$, где k не содержит неизвестных текущей задачи. Переменная c идентифицируется с суммой всех известных слагаемых показателя при a, после чего определяется невырожденная сумма b.

(l) Занесение под конечную сумму дополнительного слагаемого.

$$\forall_{abfmn}\left(a=f(m-1)\rightarrow\sum_{i=m}^{n}f(i)+a+b=\sum_{i=m-1}^{n}f(i)+b\right)$$

Антецедент, выделенный указателем "идентификатор", сравнивает выражение a с результатом упрощения выражения $f(m-1)$.

(m) Выражения с "O". При применении нормализатора к выражениям с "O" применяется следующий прием, поглощающий меньшее слагаемое:

$\forall_{mnxa}((x\rightarrow 0\backslash a)\ \&\ m\leq n\rightarrow O(x^m)+O(x^n)=O(x^m))$

Здесь m,n идентифицируются с натуральными константами; антецедент представляет собой фиктивную посылку, указывающую контекст "стремления" к нулю переменной x.

6. Тождества, преобразующие к виду суммы. Собственно "раскрывание скобок" выполняется следующими приемами:

(a) Внесение минуса под знак суммы.

$\forall_{ab}(-a-b=-(a+b))$

Заголовок приема - "замена(первыйтерм стандплюс)".

(b) Использование дистрибутивности.

$\forall_{ab}(ab+ac=a(b+c))$

Тождество применяется справа налево; указатель "набор(первыйтерм)" обеспечивает одновременное рассмотрение всех слагаемых. Каждое новое произведение сначала обрабатывается нормализатором "нормумножение", а затем - нормализатором "стандплюс". Таким образом, раскрывание скобок в данном подвыражении доводится до конца. Результирующая сумма обрабатывается нормализатором "нормплюс", устраняющим вложенные суммы и подобные члены.

(c) Квадрат суммы.

$\forall_{bc}((b+c)^2=2bc+b^2+c^2)$

Удвоенное произведение и квадраты слагаемых обрабатываются сначала нормализатором общей стандартизации, а затем - нормализатором "стандплюс".

(d) Куб суммы.

$\forall_{bc}((b+c)^3=3bc^2+3b^2c+b^3+c^3)$

(e) Понижение степени суммы. Начиная с четвертой степени, используется пошаговое понижение показателя. Для ускорения вычислений здесь используются формулы квадрата и куба суммы; первая - для показателей 4 и 5, вторая - для больших 5:

$\forall_{abc}(0<a-3\ \&\ a-6<0\rightarrow(b+c)^a=2bc(b+c)^{a-2}+b^2(b+c)^{a-2}+c^2(b+c)^{a-2})$

$\forall_{abc}(0<a-5\rightarrow(b+c)^a=3bc^2(b+c)^{a-3}+3b^2c(b+c)^{a-3}+b^3(b+c)^{a-3}+c^3(b+c)^{a-3})$

Как и выше, каждое слагаемое в заменяющем терме обрабатывается нормализатором "стандплюс".

(f) Преобразование дроби с суммой в числителе к виду суммы дробей.

$\forall_{abc}(c - \text{число} \to a/c + b/c = (a+b)/c)$

Тождество применяется справа налево; указатель "набор(первыйтерм)" обеспечивает одновременное рассмотрение всех слагаемых числителя. Обращения к нормализаторам "нормдробь", "стандплюс" введены только для первого слагаемого заменяющего терма. Так как компилятор организует цикл обработки слагаемых, то фактически нормализаторы будут применяться к каждому из них.

7. Преобразование к виду суммы тригонометрических выражений.

Если при обращении к нормализатору был введен комментарий "видумножение", то дополнительно происходит преобразование произведений тригонометрических выражений в суммы. Для этого используются следующие приемы:

(a) Произведение синусов.

$$\forall_{abcd}\left(c = a - b\ \&\ d = a + b \to \sin a \sin b = \frac{\cos c}{2} - \frac{\cos d}{2}\right)$$

Проверяется, что аргументы a, b различны.

(b) Произведение синуса на косинус.

$$\forall_{abcd}\left(c = a - b\ \&\ d = a + b \to \sin a \cos b = \frac{\sin d}{2} + \frac{\sin c}{2}\right)$$

Если перемножаются синус и косинус общего аргумента, то переход к синусу двойного аргумента, уже встречающемуся в преобразуемой сумме, может происходить и без комментария "видумножение":

$$\forall_{ab}\left(b = 2a \to \sin a \cos a = \frac{\sin b}{2}\right)$$

(c) Произведение косинусов.

$$\forall_{abcd}\left(c = a - b\ \&\ d = a + b \to \cos a \cos b = \frac{\cos c}{2} + \frac{\cos d}{2}\right)$$

Проверяется, что аргументы a, b различны.

(d) Квадрат синуса.

$$\forall_{abc}\left(a = 2b \to (\sin c)^a = \frac{(1 - \cos(2c))^b}{2^b}\right)$$

Переменная a идентифицируется с натуральной константой. Антецедент выделен указателем "программа", т.е. будет проверяться, что a - четное, и определяться частное b от деления a на 2.

(e) Квадрат косинуса.

$$\forall_{abc}\left(a = 2b \to (\cos c)^a = \frac{(1 + \cos(2c))^b}{2^b}\right)$$

(f) Куб синуса.

$$\forall_{ab}\left(a = 3b \to (\sin b)^3 = \frac{3\sin b - \sin a}{4}\right)$$

(g) Куб косинуса.

$$\forall_{ab}\left(a = 3b \to (\cos b)^3 = \frac{3\cos b + \cos a}{4}\right)$$

(h) Понижение степени синуса.

$$\forall_{ab}\left((\sin a)^b = \frac{(\sin a)^{b-2}}{2} - \frac{(\sin a)^{b-2}\cos(2a)}{2}\right)$$

Переменная b идентифицируется с натуральной константой, большей трех. Уровень срабатывания этого приема равен 2, в то время как у предыдущих тригонометрических приемов он был равен 1.

(i) Понижение степени косинуса.

$$\forall_{ab}\left((\cos a)^b = \frac{(\cos a)^{b-2}}{2} + \frac{(\cos a)^{b-2}\cos(2a)}{2}\right)$$

8. Преобразование к виду суммы гиперболических функций.

Комментарий "видумножение" инициирует также преобразование произведений гиперболических функций в суммы:

(a) Квадрат гиперболического косинуса.

$$\forall_{abc}\left(a = 2b \to (\operatorname{ch} c)^a = \frac{(1 + \operatorname{ch}(2c))^b}{2^b}\right)$$

(b) Квадрат гиперболического синуса.

$$\forall_{abc}\left(a = 2b \to (\operatorname{sh} c)^a = \frac{(\operatorname{ch}(2c) - 1)^b}{2^b}\right)$$

(c) Произведение гиперболических синусов.

$$\forall_{abcd}\left(c = a - b \ \&\ d = a + b \to \operatorname{sh} a \operatorname{sh} b = \frac{\operatorname{ch} d}{2} - \frac{\operatorname{ch} c}{2}\right)$$

(d) Произведение гиперболических косинусов.

$$\forall_{abcd}\left(c = a - b \ \&\ d = a + b \to \operatorname{sh} a \operatorname{sh} b = \frac{\operatorname{ch} c}{2} + \frac{\operatorname{ch} d}{2}\right)$$

(e) произведение гиперболического синуса на косинус.

$$\forall_{abcd}\left(c = a - b \ \&\ d = a + b \to \operatorname{sh} a \operatorname{ch} b = \frac{\operatorname{sh} d}{2} + \frac{\operatorname{sh} c}{2}\right)$$

Нормализатор стандартизации сумм с неизвестными "уравнплюс"

Это нормализатор из той же серии, что рассмотренный выше нормализатор "уравнминус". Он имеет следующие приемы:

1. Приведение подобных членов с неизвестными.

   $\forall_{abc}(ab + ac = a(b + c))$

   Переменная a идентифицируется с помощью оператора "алгебрпересечение" как общая часть множителей двух слагаемых. Она содержит неизвестные. Остаточные части b, c известны. Ускоряющий фильтр "постпозиция(фикс(0 1 2) фикс(0 1 1))" отсекает рассмотрение симметричных случаев, отличающихся перестановкой слагаемых: второе слагаемое должно идти в сумме после первого. Ускоряющий фильтр "подобныечлены(фикс(0 1 1)фикс(0 1 2)неизвестные)" предпринимает попытку отсечь заведомо не подобные члены до обращения к оператору "алгебрпересечение". Сумма $b + c$ обрабатывается нормализаторами "нормплюс" и "стандплюс". Иногда такая стандартизация позволяет сильно упростить коэффициент.

2. Сложение подобных дробей с неизвестными.

   $$\forall_{abcdpq}\left(\frac{ap}{bq} + \frac{cp}{dq} = \frac{p(ad + bc)}{bdq}\right)$$

   Общая часть p/q содержит неизвестные; коэффициенты a, b, c, d - не содержат. Сумма в числителе результирующей дроби обрабатывается нормализаторами "нормплюс", "видумножение".

3. Рекурсивное обращение к стандартизации слагаемых. Нормализаторы стандартизации подвыражений с неизвестными - корневые. Если нужно обеспечить сквозную стандартизацию подвыражений с неизвестными, при обращении к этим нормализаторам вводится комментарий "нормуравн". Тогда применяются приемы рекурсивной обработки операндов. Для суммы имеем:

   $\forall_{abc}(c = a \to a + b = b + c)$

   Антецедент обращается к обработке слагаемого a процедурой "нормуравн", которая организует обращение к подходящему нормализатору выражений с неизвестными, определяемому по заголовку слагаемого. Проверяется отличие выражения c от a.

4. Уменьшение числа тригонометрических операций с неизвестным аргументом. Используются приемы преобразования суммы тригонометрических выражений в произведение:

   $\forall_{bcpq}(p = b + c \ \& \ q = b - c \to \sin b - \sin c = 2\cos(p/2)\sin(q/2))$

   $\forall_{bcpq}(p = b + c \ \& \ q = b - c \to d\sin b + d\sin c = 2d\sin(p/2)\cos(q/2))$

   $\forall_{bcpq}(p = b + c \ \& \ q = b - c \to d\cos b + d\cos c = 2d\cos(p/2)\cos(q/2))$

   $\forall_{bcpq}(p = b + c \ \& \ q = b - c \to \cos b - \cos c = -2\sin(p/2)\sin(q/2))$

   Во всех этих случаях проверяется, что хотя бы одно из выражений p, q после обработки нормализатором "нормплюс" оказывается не содержащим неизвестных. При этом b - сумма, содержащая неизвестные. Коэффициент d может обращаться в единицу либо иметь знак "минус". В качестве упражнения рекомендуется ввести такой коэффициент в первый и четвертый приемы.

Нормализатор группировки "групплюс"

Это очень простой нормализатор, выполняющий приведение подобных членов относительно подвыражений, содержащих заданные переменные. Последние задаются с помощью передаваемых нормализатору комментариев (переменная x). Нормализатор имеет единственный прием:

$\forall_{abc}(ab + ac = a(b + c))$

Указатель "выписка(x1 контекст(комментарий(переменная переменная(x4))входит(x4 x1)))"определяет a как произведение всех общих множителей двух слагаемых, содержащих выделенные переменные. Фильтр "не(контекст(комментарий(переменная переменная(x4))или(входит(x4 x2)входит(x4 x3))))" проверяет, что остаточные множители b, c не содержат выделенных переменных.

Анализатор "сложениеуравнений"

При решении систем уравнений сначала применяются эквивалентные преобразования. Однако, если они не приводят к получению ответа, то на уровне 4 предпринимается обращение к сканированию блока анализа - вспомогательной задачи на исследование, в которой собраны вместе уравнения и условия на известные параметры задачи. Здесь реализуется вывод следствий из уравнений, ориентированный на получение значений неизвестных. Не все утверждения, получаемые при выводе следствий, равноценны. Однако, сразу дать адекватную оценку бывает трудно; иногда внешне малопримечательное следствие через два-три шага приводит к получению ключевого утверждения. Поэтому приходится сохранять в списке посылок все выводимые следствия, и при накоплении достаточно большого их количества ход решения существенно замедляется.

Для борьбы с данным явлением и были созданы пакетные анализаторы. Они обеспечивают вывод большого числа следствий вне рамок сканирования задачи, с привлечением лишь малого числа своих внутренних приемов. При выводе происходит отбор наиболее ценных утверждений: они помечаются комментарием "внешвывод". По окончании работы анализатора, обычно наступающему ввиду исчерпания отведенного лимита трудоемкости, отобранные утверждения перемещаются в посылки текущей задачи. Таким образом, все промежуточные утверждения, ценность которых не очевидна, выводятся из трудоемкого процесса сканирования задачи. В особых случаях работа анализатора, нашедшего ценное утверждение, может быть оборвана немедленно.

Анализатор "сложениеуравнений", главным образом, обеспечивает порождение линейных комбинаций уравнений. Вне него линейные комбинации уравнений при выводе следствий обычно не порождаются. Приемы, выполняющие обращение к данному анализатору, были приведены выше. Перечислим приемы анализатора :

1. Линейная комбинация уравнений, имеющих неизвестные члены, отличающиеся лишь известными коэффициентами.

   $\forall_{abcdefgh}(ag + b = c \;\&\; ah + d = e \;\&\; f = bh - dg - ch + eg \to f = 0)$

   Прием имеет заголовок "внутрвывод(сложениеуравнений)". Первые два антецедента идентифицируются с уравнениями. Указатель "перечень(x7 известно(x7))" определяет идентификацию g с произведением всех известных сомножителей текущего члена первого уравнения. Оставшиеся сомножители дают a; проверяется, что a имеет неизвестные (т.е. невырождено). Проверяется, что выражение h, определяемое путем отбрасывания множителей a в текущем члене второго уравнения, не содержит неизвестных. Проверяется также отсутствие дробных слагаемых уравнений (предполагается, что до обращения к анализатору их должны были устранить другие приемы). Третий антецедент, выделенный указателем "идентификатор", определяет левую часть f линейной комбинации уравнений. Она обрабатывается нормализаторами "нормплюс", "стандплюс", "уравнплюс" и "факторизация". Результирующее уравнение $f = 0$ обрабатывается нормализатором "нормчисло".

Проверяется, что выведенное уравнение лучше исходных: либо имеет меньше неизвестных слагаемых, чем каждое из них, либо имеет меньше неизвестных. При наличии комментария "полный", указывающего на усиленный режим, результирующее уравнение допускается, если оно линейно хотя бы по одной неизвестной.

Обратим внимание на фильтры, проверяющие эти условия. Первый из них имеет вид "упрощение(x9 и(x6 фикс(1 1)фикс(2 1))число(вид(x9 плюс(x10 x11)) единица(0 x11)не(известно(x10))))". Такая конструкция, использующая символ "упрощение", встречается достаточно редко. Переменная x9 последовательно идентифицируется с x6 (т.е. f), с левой частью первого уравнения и левой частью второго (указатели "фикс(1 1)" и "фикс(2 1)"). Для нее определяется значение выражения "число(...)", т.е. число ее неизвестных слагаемых. Так как альтернативы x6, "фикс(1 1)", "фикс(2 1)" соединены связкой "и", проверяется, что значение, найденное для x6, меньше каждого из остальных значений. В случае связки "или" проверялось бы, что значение для x6 меньше хотя бы одного из них.

Перейдем к рассмотрению указателей, определяющих отбор ценных следствий и обрыв вывода. Указатель "обрыв(равно(числонеизвестных(x6)1))" распознает случай, когда новое уравнение имеет единственную неизвестную. Тогда оно помечается комментарием "внешвывод" для последующего перенесения в список посылок текущей задачи, причем работа анализатора обрывается. Аналогичный указатель обрывает работу анализатора, если f есть сумма двух степеней неизвестных, имеющих известные коиффициенты и одинаковый известный показатель. Указатель "обрыв(заголовок(результат или))" обрывает работу, если после обработки нормализатором "нормчисло" уравнение $f = 0$ оказалось декомпозировано - распалось в дизъюнкцию других уравнений.

Указатель "примечание(условие(и(не(контекст(неизвестная(x9)список(x10 фикс(1 1)фикс(2 1))линейно(x10 x9)))контекст(неизвестная(x9)линейно(x6 x9))) внешвывод))" помечает новое уравнение комментарием "внешвывод", если оно линейно по некоторой неизвестной, а оба исходных уравнения - не линейны ни по какой своей неизвестной. Аналогичный комментарий помечает уравнение, если оно линейно по всем неизвестным задачи, а хотя бы одно из исходных - нелинейно по какой-либо неизвестной. В обоих случаях работа анализатора после этого продолжается.

Заметим, что уровень срабатывания приема анализатора (как и нормализатора) определяется не фильтром "уровень(...)", а указателем такого же вида, т.е. помещается в 4-м окне редактора ГЕНОЛОГа. Если указателя нет (как в нашем случае), то уровень срабатывания равен 1.

2. Усмотрение степени линейной комбинации двух неизвестных в левой части уравнения.

Прием рассматривает сумму всех неизвестных слагаемых левой части уравнения; пытается разложить ее на множители, и если результат имеет вид степени суммы линейной комбинации двух неизвестных, то выводится уравнение с этой степенью в левой части:

$\forall_{abcdenxy}(a = (dx + ey)^n \ \& \ a + b = c \to (dx + ey)^n = c - b)$

Второй антецедент идентифицируется с уравнением. b определяется как сумма всех известных слагаемых одной его части; противоположная часть c известна. Остаточная сумма a невырождена; первый антецедент обращается для ее преобразования к нормализатору разложения на множители "видумножение". Результат идентифицируется со степенью $(dx + ey)^n$. При этом n - натуральная константа; x, y - неизвестные; d, e - известные выражения. Уровень срабатывания равен 1. Прием имеет указатель "обрыв", т.е. при его срабатывании результирующее уравнение помечается комментарием "внешвывод" и работа анализатора обрывается. Во внешней задаче найденное уравнение будет преобразовано к линейному уравнению относительно x, y.

3. Линейная комбинация уравнений в случае неизвестных коэффициентов.

   Имеются два приема, отличающихся уровнями срабатывания. Теорема их та же, что для случая известных коэффициентов:

   $\forall_{abcdefgh}(ag + b = c \ \& \ ah + d = e \ \& \ f = bh - dg - ch + eg \to f = 0)$

   Первый прием срабатывает на уровне 1. Выражение a идентифицируется как произведение всех общих множителей текущих членов уравнений; здесь применяется оператор "алгебрпересечение". Оно содержит неизвестные. Хотя бы один из коэффициентов g, h не известен. Уравнения не имеют дробных слагаемых; линейная комбинация f отлична от нуля. Требуется выполнение следующих условий:

   (a) Либо число вхождений неизвестных в f меньше, чем в каждое из исходных уравнений, либо число слагаемых в f меньше, чем хотя бы в одном из исходных уравнений, либо число неизвестных в f равно 1, в каждом из исходных уравнений - больше единицы, причем a содержит тригонометрическую операцию;

   (b) Общее число вхождений в уравнения дробных степеней с неизвестным основанием не равно 1 (иначе такие степени не смогут сократиться при получении линейной комбинации);

   (c) Если g не известно, то либо b линейно, либо b имеет слагаемое, делящееся на g, либо существует неизвестная, входящая в a и не входящая в b, g. Симметричное условие выполняется для h.

   Обрыв работы анализатора происходит в следующих случаях: f имеет слагаемое kx, где k известно, а x не входит в остальные слагаемые; результирующее уравнение после нормализации превратилось в дизъюнкцию; результирующее уравнение линейно по всем своим неизвестным; f имеет вид линейной комбинации суммы двух степеней неизвестных с равными показателями; единственное неизвестное слагаемое выражения f имеет вид kp^q, где k, p известны. При отсутствии обрыва новое уравнение помечается комментарием "внешвывод", если оно линейно по какой-либо неизвестной, а оба исходных уравнения - нелинейны ни по какой неизвестной.

Второй прием срабатывает на уровне 2. В этом случае a - десятичная константа. Выражение f обрабатывается нормализатором "видумножение". Проверяется, что его заголовком, при отбрасывании минуса, является "умножение" либо "степень". Если результирующее уравнение преобразуется в дизъюнкцию, происходит обрыв работы анализатора.

4. Попытка разложить на множители левую часть линейной комбинации уравнений в случае известных коэффициентов.

   Усиленный нормализатор разложения на множители "видумножение" применялся в рассмотренных выше приемах к линейной комбинации уравнений только в одном случае - для приема уровня 4. В остальных случаях применялся ослабленный нормализатор "факторизация". Для линейной комбинации с известными коэффициентами создана версия приема, использующая нормализатор "видумножение". Она срабатывает на уровне 2 при выполнении следующих условий: a содержит неизвестные, коэффициенты g и h известны, отсутствуют дробные слагаемые, а f после отбрасывания известных множителей и минуса имеет заголовок "умножение" либо "степень". Обрыв работы анализатора происходит, если новое уравнение имеет единственную неизвестную, либо преобразуется в дизъюнкцию, либо f является линейной комбинацией двух неизвестных. При отсутствии обрыва комментарий "внешвывод" вводится, если новое уравнение линейно по какой-либо неизвестной, а оба исходных уравнения - нет.

Синтезатор деления многочленов "частноемногочленов"

Синтезатор предназначен для деления с остатком двух многочленов от одной вещественной переменной, заданных в виде функциональных термов $\lambda_x(f(x), x - \text{число})$. В отличие от формальных многочленов, рассматриваемых в последующих разделах, такие термы определяют соответствующие многочленам функции. Использоваться синтезатор будет при преобразованиях к виду сумм простейших дробей. Напомним, что пакетные синтезаторы аналогичны задачам на описание: они подбирают такие значения выходных переменных, при которых становится истинным реализуемое синтезатором утверждение. В нашем случае реализуется утверждение "частноемногочленов(a b c d)", где a, b - делимое и делитель; c, d - неполное частное и остаток. Синтезатор имеет три приема:

1. Степень делимого меньше степени делителя.

   В этом случае сразу определяется нулевое неполное частное и делимое выдается в качестве остатка:

   $\forall_{abcdefgh}(a = \lambda_x(bx^c + f(x), x - \text{число})\ \&\ d = \lambda_x(ex^g + h(x), x - \text{число})\ \&\ 0 < g - c \rightarrow \text{частноемногочленов}(a, d, \lambda_x(0, x - \text{число}), a))$

   Первые два антецедента, выделенные указателем "идентификатор", обеспечивают идентификацию делимого и делителя как функциональных термов, определяющих многочлены. При этом идентифицируются главные члены bx^c, ex^g и суммы младших членов $f(x), h(x)$. Чтобы задать компилятору режим идентификации многочленов, введены указатели "смнногочлен(x1 x23)", "сммногочлен(x4 x23)", "сммногочлен(x6 x23)", "сммногочлен(x8 x23)". Первые два нужны для обращения к процедуре "сммногочлен", определяющей набор коэффициентов многочлена; последние два - для выделения главного члена.

2. Степень делимого не меньше степени делителя.

   Выполняется обычный шаг деления многочленов "столбиком": делитель домножается на частное главных членов и вычитается из делимого.

   $\forall_{abcdefghiju}(a = \lambda_x(bx^c + f(x), x - \text{число})\ \&\ d = \lambda_x(ex^g + h(x), x - \text{число})\ \&\ i = c - g\ \&\ 0 \leq i\ \&\ \neg(e = 0)\ \&\ \text{частноемногочленов}(\lambda_x(f(x) - (bx^i h(x))/e, x - \text{число}), d, u, j) \to \text{частноемногочленов}(a, d, \lambda_x((bx^i)/e + u(x), x - \text{число}), j))$

   Идентификация многочленов выполняется первыми двумя антецедентами аналогично предыдущему случаю. Шестой антецедент реализует рекурсивное обращение для завершения деления. Чтобы компилятор понимал, что u возвращается в виде функционального терма $\lambda_y(P(y), y - \text{число})$, и мог получить $u(x)$ путем подстановки x вместо y в $P(y)$, введен указатель "функция(x20 x23)".

3. Деление двух константных многочленов.

   $\forall_{ab}(\neg(b = 0) \to \text{частноемногочленов}(\lambda_x(a, x - \text{число}), \lambda_x(b, x - \text{число}), \lambda_x(a/b, x - \text{число}), \lambda_x(0, x - \text{число})))$

   Фактически прием применяется только для деления нулевого многочлена на константный.

Далее перечисляются отнесенные к символу "плюс" пакетные операторы, введенные для узкоспециальных целей.

Нормализатор "Норммногочлен" преобразования выражения к виду многочлена

Этот нормализатор используется в тех случаях, когда нужно преобразовать выражение к стандартному виду многочлена относительно одной из переменных x, явно выделенной в комментарии "переменная x". Например, к нему происходят интенсивные обращения при нахождении конечного числа членов разложения в ряд Тейлора. Нормализатор корневой. Его приемы обеспечивают раскрывание скобок и группировку членов по степеням выделенной переменной:

1. Внесение минуса под знак суммы.

2. Приведение подобных членов.

   Прием учитывает возможность появления дробных слагаемых:

   $$\forall_{abcdxn}\left(\frac{ax^n}{c} + \frac{bx^n}{d} = \left(\frac{a}{c} + \frac{b}{d}\right)x^n\right)$$

   Здесь x - выделенная переменная; a, b, c, d не содержат x; n - натуральная константа. Знаменатели могут вырождаться.

3. Сложение с нулем.

4. Раскрывание скобок. Используются приемы, аналогичные рассматривавшимся ранее, но так как нормализатор корневой, то все преобразования привязаны к его корневой операции. Например, появляется прием $\forall_{abcd}(a(b + c) + d = ab + ac + d)$, учитывающий возможность наличия внешних слагаемых d для

преобрузуемого произведения $a(b+c)$. При возведении суммы в степень используются два приема:

$\forall_{abnk}(n = 2k \rightarrow (a+b)^n = (a+b)^k(a+b)^k)$

$\forall_{abn}((a+b)^n = (a+b)(a+b)^{n-1})$

Первый уменьшает четный показатель степень вдвое; второй - сводит нечетный показатель к четному (его уровень срабатывания больше, чем у первого).

5. Отбрасывание высших степеней.

   $\forall_{abcxn}((ax^n)/c + b = b)$

   Прием применяется, если имеется комментарий (степень k), указывающий на отбрасывание степеней x, больших k, причем n - натуральная константа, большая k. Такая ситуация возникает при разложении выражения в ряд Тейлора до заданного числа членов.

6. Преобразование дроби с суммой в числителе к виду суммы дробей.

   $$\forall_{abcde}\left(\frac{e(a+b)}{c} + d = \frac{ae}{c} + \frac{be}{c} + d\right)$$

   Выделенная переменная x входит в сумму $a+b$.

7. Устранение вложенных сумм.

Нормализатор "нормкоэфф" построения многочлена по заданному набору коэффициентов

Нормализатор преобразует выражение "коэфф(a x)", где a - набор коэффициентов, в сумму степеней выражения x с данными коэффициентами. Фактически он использован в решателе лишь однажды - при интегрировании функций специального вида. Использование записи "коэфф(...)" позволяет избежать более громоздкой записи через конечную сумму и уменьшает текст теоремы. Приведем прием нормализатора:

$$\forall_{anx}\left(l(a) = n \rightarrow \text{коэфф}(a,x) = \sum_{i=1}^{n} a(i)x^{i-1}\right)$$

Конечная сумма выделена указателем "развертка"; величина n определяется антецедентом с помощью нормализатора "нормдлинанабора". Чтобы в результирующую сумму сразу подставлялись соответствующие коэффициенты, выражение $a(i)$ обрабатывается нормализатором "нормзначение".

Нормализатор "умно箭дробь" предварительного раскрывания скобок

Нормализатор усматривает возможность сокращений при раскрывании скобок с дробными слагаемыми. Он используется в аналитической геометрии при стандартизации уравнения кривой либо поверхности. Приемы нормализатора таковы:

$$\forall_{abcd}\left(a + b\left(\frac{c}{b} + d\right) = a + c + bd\right)$$

$$\forall_{abcd}\left(a + b\left(\frac{c}{\sqrt{b}} + d\right) = a + c\sqrt{b} + bd\right)$$

$$\forall_{abcde}\left(a+be\left(\frac{c}{b}+d\right)=a+ce+bde\right)$$

$$\forall_{abcde}\left(a+be\left(\frac{c}{b\sqrt{e}}+d\right)=a+c\sqrt{e}+bed\right)$$

$$\forall_{abcde}\left(a+be\left(\frac{c}{\sqrt{b}\sqrt{e}}+d\right)=a+c\sqrt{b}\sqrt{e}+bed\right)$$

$$\forall_{abcde}\left(a+be\left(\frac{c}{\sqrt{b}}+d\right)=a+ce\sqrt{b}+bde\right)$$

Синтезатор "смкоэфф" определения коэффициента при заданном члене, возникающего после раскрывания скобок

Синтезатор используется в теории вероятностей при рассмотрении повторения опытов, имеющих несколько исходов. Он позволяет определить коэффициент одночлена с заданным набором степеней переменных, возникающего при раскрывании скобок в произведении линейных двучленов. При этом явного выписывания результата раскрывания скобок не происходит.

Используемая в синтезаторе запись "смкоэфф(a, x, n, b)" означает, что a - выражение, в котором требуется раскрыть скобки; x - набор переменных; n - набор степеней этих переменных; b - искомый коэффициент. Заметим, что эта запись является лишь техническим сокращением. Чтобы она стала логически корректной, нужно было бы рассматривать формальный многочлен, а не выражение a для вычисления его значений. Однако, тогда при вычислениях использовались бы значительно более громоздкие структуры данных. Такого рода формальные неточности при программировании на ГЕНОЛОГе вполне оправданы, так как возникают в легко контролируемых контекстах небольших пакетных операторов и гарантируют получение правильного ответа. Необходимые для логической корректности компоненты записей, не участвующие в вычислениях, здесь просто "выносятся за скобки".

Приведем прием, обеспечивающий рекурсивное определение коэффициента при рассмотрении произведения линейных двучленов:

$\forall_{abcixmN}(l(x) = n\ \&\ i \in \{1, \dots, n\}\ \&\ m = N(i)\ \&\ 0 < m\ \&\ \text{смкоэфф}(ac, x, \lambda_j((m - 1\text{при}j = i, \text{иначе}N(j)), j \in \{1, \dots, n\}), p)\ \&\ \text{смкоэфф}(bc, x, N, q)\ \&\ r = (ax(i) + b)c \rightarrow \text{смкоэфф}(r, x, N, p + q))$

Первый антецедент определяет число переменных n. Второй, выделенный указателем "программа", выбирает произвольный номер переменной i. Третий антецедент, выделенный указателем "идентификатор", определяет степень m при переменной $x(i)$ в искомом одночлене. Следующий антецедент, выделенный указателем "блок-проверок", проверяет, что эта степень положительная. Последний антецедент выделяет в произведении двучленов множитель $(ax(i) + b)$. Два предыдущих антецедента реализуют рекурсивные обращения к раскрыванию скобок в выражениях ac и bc. Синтезатор имеет два сопровождающих приема:

$\forall_{abx}(\text{смкоэфф}(b, x, a, b))$

$\forall_{aximN}(l(x) = n\ \&\ i \in \{1, \dots, n\}\ \&\ 0 < m\ \&\ N(i) = m \rightarrow \text{смкоэфф}(a, x, N, 0))$

Они нужны для обработки заключительных ситуаций: константного выражения при нулевом наборе степеней переменных, и усмотрения нулевого коэффициента, если какая-либо переменная одночлена не входит в преобразуемое выражение.

Нормализатор "стандсумма" преобразования выражения к виду многочлена относительно заданного списка переменных

Нормализатор применяется при преобразовании квадратичной формы к нормальному виду. Он выполняет простейшие вспомогательные преобразования: приводит подобные члены относительно множителей, содержащих выделенные переменные, а также устраняет вложенные суммы и нулевые слагаемые.

10.6 Приемы символа "умножение"

Выражение "умножение($a_1 \dots a_n$)", прорисовываемое формульным редактором в виде $a_1 \dots a_n$, обозначает произведение чисел $a_1, \dots, a_n$. Справочники "арность", "ассоциативно", "коммутативно", "тип", "одз", "типданных", "область" указывают, что операция "умножение" имеет два операнда, ассоциативна, коммутативна, определена на вещественных числах и принимает числовые значения, причем любое такое значение. Справочник "единица" дает ее единицу; справочник "заменазнака" указывает, что можно выносить за знак операции минус перед сомножителем. Справочник "суммавсех" определяет символ "произведениевсех", используемый для обозначения произведения конечного набора чисел. Справочники "конст" и "ноль" независимо друг от друга указывают на ноль операции (справочник "конст" дает значение операнда, обращающее всю операцию в константу). Справочник "дистрибутивно" проверяет, что умножение дистрибутивно относительно сложения; справочник "дистрибразвертка" - указывает на сложение как на операцию, относительно которой умножение дистрибутивно. Используемый компилятором справочник "вычисл" определяет операторы ЛОСа для вычисления произведений в различных типах данных, а также для деления. Справочник "степень" указывает, что произведение конечного числа одинаковых сомножителей выражается через операцию "степень". Справочники "неизвоценка", "сокращение", "подстановка", "заголовок" используются при выводе теорем. Справочник "монотонно" указывает условия нестрогой монотонности произведения относительно сомножителей.

Простейшая стандартизация

Приемы устранения вложенных операций и лексикографического упорядочения операндов - такие же, как в случае сложения. Для умножения десятичных чисел имеется прием, использующий реализованный на ЛОСе нормализатор "умножить". Единичные сомножители исключаются приемом $\forall_a (a \cdot 1 = a)$, нулевые - приемом $\forall_a (a \cdot 0 = 0)$. Оба приема применяются на уровне 0 ко всем термам задачи, включая сопровождающие по о.д.з. Минус перед сомножителем выносится за знак умножения приемом $\forall_{ab} (-ab = a(-b))$. Он тоже имеет нулевой уровень срабатывания и применяется (справа налево) без ограничений.

Одновременное изменение знака у двух сомножителей, имеющих вид разности

$\forall_{abcde} (e - \text{rational} \ \& \ \neg(\text{знаменатель}(e) - \text{even}) \rightarrow (a-b)^e (c-d)^e = (b-a)^e (d-c)^e)$

Прием применяется в условии задачи на преобразование, если заменяющий терм лексикографически предшествует заменяемому. Он устраняет двойственность в записи одного и того же значения, что может оказаться необходимым для срабатывания различных приемов (например, для сокращения дробей). Преобразуемое произведение не является корневым а также не является слагаемым либо операндом уравнения или неравенства. Задача не имеет цели "разложитьнамножители". Прием блокируется при редактировании чрезмерно длинного ответа. Уровень срабатывания его равен 3.

Усмотрение разности квадратов

В ряде случаев произведение суммы двух выражений на их разность целесообразно представить как разность квадратов. Для этого преобразования используются следующие приемы:

$\forall_{abcd}(d = a^2 - b^2 \ \& \ c - \text{rational} \ \& \ \neg(\text{знаменатель}(c) - \text{even}) \to (a-b)^c(a+b)^c = d^c)$

$\forall_{abcd}(d = a^2 - b^2 \ \& \ 0 \le a - b \ \& \ 0 \le a + b \to (a-b)^c(a+b)^c = d^c)$

Наличие двух приемов вместо одного объясняется распадением о.д.з. для степени на два подкласса: степени, имеющие рациональный показатель с нечетным знаменателем, и степени с неотрицательным основанием. Это - типичная ситуация для приемов со степенью в элементарной алгебре. Чтобы срабатывание второго приема не нарушало сопровождения по о.д.з., он снабжен указателем "вывод(меньшеилиравно(0 x4))", регистрирующим в посылках условие неотрицательности нового основания степени d.

Преобразуемое произведение входит в условие задачи на преобразование, не имеющей цели "разложитьнамножители". Если оно является основанием степени, то уровень срабатывания равен 2, иначе - равен 1. Первый антецедент выполняет упрощение разности квадратов d, применяя для этого нормализаторы "нормплюс" и "стандплюс".

Для уточнения целесообразности перехода к разности квадратов введен пакетный оператор фильтра "фильтрквадратов", применяемый обоими приемами. Этот оператор, реализованный на ГЕНОЛОГе, истинен в следующих ситуациях:

1. Выражения $(a-b)^c$, $(a+b)^c$ суть множители числителя либо знаменателя дроби, причем противоположный операнд дроби делится на d.
2. Слагаемые a, b - константы, хотя бы одна из которых имеет вид радикала четной степени.
3. Рассматриваемое произведение расположено в корне условия задачи.
4. Хотя бы одно из выражений $(a-b)^c$, $(a+b)^c$ представляет собой радикал четной степени n, причем вне этого выражения не встречается радикал четной степени, делящейся на n, имеющий то же самое основание.

Указатели "замечание(Стандплюс x4 плюс(x1 x2))" и "замечание(Слагаемое x4)" блокируют обратное преобразование.

Фильтры "не(цель(нормИнтеграл))" блокируют применение данных приемов, так как для преобразования подынтегрального выражения при формальном интегрировании созданы другие два приема, основанные на тех же теоремах. Они применяются в задачах на преобразование, имеющих цель "нормИнтеграл" и комментарий (нормИнтеграл x), указывающий переменную интегрирования x. Оператор "фильтрквадратов" здесь не используется. Вместо этого имеется фильтр, ориентированный на достижение некоторых стандартных "шаблонов интегрирования". Требуется, чтобы переменная интегрирования входила в $a+b$, причем либо c представляло собой дробь, либо все произведение являлось основанием степени с дробным показателем. В последнем случае, если знаменатель дроби четный, выражение $a+b$ не должно встречаться вне рассматриваемого произведения. Добавлен еще один аналогичный прием, в котором сумма и разность заключены под модулями:

$\forall_{abcd}(d = a^2 - b^2 \rightarrow |a-b|^c|a+b|^c = |d|^c)$

Все эти приемы интегрирования срабатывают на уровне 5. В разделе имеются два простых приема, основанные на последовательном применении перехода от произведения к разности квадратов и обратного перехода. Они позволяют избавиться от модуля:

$\forall_{abc}((ab + c|b|)(ab - c|b|) = b^2(a+c)(a-c))$

$\forall_{abc}((ab + c|b|)(-ab + c|b|) = b^2(a+c)(c-a))$

Уровень срабатывания равен 0. Наконец, имеется прием, преобразующий произведение суммы двух константных выражений на их разность к виду разности квадратов. Он применяется в посылках задач на доказательство либо исследование и имеет уровень срабатывания 0.

Усмотрение разности либо суммы кубов

Раздел аналогичен предыдущему. Сначала идут приемы, основанные на следующих двух теоремах:

$\forall_{abcd}(d = a^3 - b^3 \ \& \ c - \mathrm{rational} \ \& \ \neg(\text{знаменатель}(c) - \mathrm{even}) \rightarrow (a-b)^c(a^2+ab+b^2)^c = d^c)$

$\forall_{abcd}(d = a^3 + b^3 \ \& \ c - \mathrm{rational} \ \& \ \neg(\text{знаменатель}(c) - \mathrm{even}) \rightarrow (a+b)^c(a^2-ab+b^2)^c = d^c)$

Специального оператора фильтра для них не создано, однако условия срабатывания аналогичны случаю разности квадратов: либо усматривается возможность сократить дробь, либо a и b - константы, хотя бы одна из которых находится под кубическим раздикалом, либо преобразуемое произведение - корневое. Уровень срабатывания равен 2. Для произведений с модулем введены следующие приемы:

$\forall_{abcd}(d = a^3 - b^3 \ \& \ c - \mathrm{rational} \ \& \ \neg(\text{знаменатель}(c) - \mathrm{even}) \rightarrow |a-b|^c(a^2+ab+b^2)^c = \mathrm{sg}(a-b)d^c)$

$\forall_{abcd}(d = a^3 + b^3 \ \& \ c - \mathrm{rational} \ \& \ \neg(\text{знаменатель}(c) - \mathrm{even}) \rightarrow |a+b|^c(a^2-ab+b^2)^c = \mathrm{sg}(a+b)d^c)$

Уровень и условия срабатывания - аналогичные (возможность сокращения дроби либо константы с кубическим радикалом). Наконец, при редактировании ответов дифференциальных уравнений рассматриваемое преобразование применяется с группировкой под общий модуль:

$\forall_{abc}(|a+b|^c|a^2 - ab + b^2|^c = |a^3 + b^3|^c)$

$\forall_{abc}(|a-b|^c|a^2 + ab + b^2|^c = |a^3 - b^3|^c)$

Умножение на условное выражение

Если одна из альтернатив условного выражения есть 0, то можно внести внешний множитель внутрь выражения, не увеличив его сложности:

$\forall_{aPb}((a \text{ при } P, \text{иначе } 0)b = (ab \text{ при } P, \text{иначе } 0))$

Прием применяется на уровне 1. Утверждение P - неравенство.

Раскрывание скобок

Приемы раскрывания скобок ориентированы на различные целевые ситуации:

1. Попытка раскрывания скобок для упрощения выражения.

   Имеются два приема, основанные на одной и той же теореме:

   $\forall_{abc}(c = a + b \to a + b = c)$.

   В обоих случаях решается задача на преобразование с целью "упростить". Антецедент обращается к нормализатору "стандплюс", выполняющему раскрывание скобок. Проверяется, что результат c короче исходного выражения. Условием применения первого приема является наличие у $a + b$ такого слагаемого, некоторый сомножитель которого есть сумма либо натуральная степень суммы. Проверяется, что данный сомножитель не играет роли коэффициента при выражении, зависящем от переменных связывающей приставки внешнего описателя "класс". В случае внешнего описателя "отображение" применение приема ограничивается еще сильнее: преобразуемое выражение должно быть заключено под сигнумом. Наконец, блокируется попытка раскрывания скобок у чрезмерно громоздкого выражения, входящего в редактируемый ответ. Второй прием применяется к сумме, содержащей произведение тригонометрических функций либо натуральную степень такой функции. Здесь обращение к нормализатору "стандплюс" сопровождается комментарием "видумножение", обеспечивающим переход от произведений тригонометрических функций к суммам. Прием блокируется, если задача имеет цель "нормИнтеграл" либо "частнпроизв". Уровень срабатывания приемов равен 3.

   На уровне 5 применяется прием, предпринимающий попытку преобразовать константное произведение тригонометрических функций в сумму для исключения тригонометрических функций:

   $\forall_{abc}(c = ab \to ab = c)$.

   Выражение a имеет своим заголовком "синус" либо "косинус"; b имеет сомножителем аналогичную тригонометрическую операцию либо ее степень. Антецедент обращается к нормализатору "стандплюс" с комментарием "видумножение". Проверяется, что c не содержит тригонометрических операций. При наличии цели "нормИнтеграл" прием блокируется. Уровень срабатывания равен 5.

   При наличии цели "раскрытьскобки" к условию задачи на преобразование применяется следующий прием:

   $\forall_{abcden}(e = a(b + c)^n + d \to a(b + c)^n + d = e)$

   Антецедент обращается к нормализатору "стандплюс". Переменная n идентифицируется с натуральной константой, возможно, равной единице. Коэффициент a тоже может обращаться в единицу, но не одновременно с n. На той же самой теореме основан еще один прием, применяемый к условию задачи на описание, имеющему вид $\neg(a(b+c)^n + d = 0)$. Эта задача должна иметь цель "пример".

2. Раскрывание скобок при интегрировании.

   Созданы два приема. Первый имеет следующую теорему:

   $$\forall_{abcdenp}\left(p = a + \frac{(b+c)^n d}{e} \rightarrow a + \frac{(b+c)^n d}{e} = p\right).$$

   Он применяется к условию задачи на преобразование, имеющей цель "нормИнтеграл". Преобразуемое вхождение суммы - корневое. Переменная интегрирования входит в $b+c$. Показатель степени n идентифицируется с натуральной константой от 1 до 7. Знаменатель e невырожденный. Если a равно 0, то d не имеет своим сомножителем функций "логарифм", "арксинус", "арккосинус", "арктангенс", "арккосинус", зависящих от переменной интегрирования. Таким образом сохраняется вид, удобный для интегрирования по частям. Кроме того, проверяется, что e не содержит тригонометрической операции, зависящей от переменной интегрирования. Антецедент обращается к нормализатору "стандплюс". Уровень срабатывания равен 5.

   Второй прием имеет следующую теорему:

   $\forall_{abcdnp}(p = a + (b+c)^n d \rightarrow a + (b+c)^n d = p)$

   Уровни срабатывания приема равны 1 и 5. Либо преобразуемая сумма является корневой, либо она является числителем дроби, знаменатель которой не зависит от переменной интегрирования, либо текущий уровень равен 5, а сумма расположена под радикалом в знаменателе дроби. В последнем случае она не должна приводиться к квадратному трехчлену от переменной интегрирования. Как и в первом приеме, введен фильтр, сохраняющий вид, удобный для интегрирования по частям.

3. Раскрывание скобок в тождествах - посылках задачи на доказательство.

   Прием выполняет вывод следствия - равенства нулю результата раскрывания скобок в разности частей тождества:

   $\forall_{abcdepq}(p = a(b+c)^d + e - q \,\&\, a(b+c)^d + e = q \rightarrow p = 0)$

   Второй антецедент идентифицируется с посылкой задачи на доказательство; первый - использует нормализатор "стандплюс" для раскрывания скобок. В геометрических задачах прием блокируется, так как раскрывание скобок выполняется там другими приемами. Показатель степени d идентифицируется с натуральной константой от 1 до 4. Уровень срабатывания равен 5.

4. Раскрывание скобок в равенствах - условиях задачи на доказательство.

   $\forall_{abcdepq}(p = a(b+c)^d + e - q \rightarrow a(b+c)^d + e = q \leftrightarrow p = 0)$

   Прием применяется к подутверждению условия задачи на доказательство. Проверяется, что результирующее равенство короче исходного. Уровень срабатывания равен 2.

5. Раскрывание скобок в выражениях с числовыми атомами.

   Прием имеет "остаточный" характер: основная часть приемов, раскрывающих скобки в выражениях со специализированными числовыми атомами (расстояния, углы и т.п.), вынесена в те разделы, где эти числовые атомы рассматриваются.

   $\forall_{abcde}(e = a(b + c) + d \rightarrow a(b + c) + d = e)$

   Прием применяется к посылке задачи на доказательство либо исследование. Сумма $b + c$ содержит символ "длина" или "вес". Уровень срабатывания равен 2.

6. Раскрывание скобок в уравнении кривой либо поверхности.

   Прием нужен для приведения к стандартному виду уравнения кривой либо поверхности, чтобы далее можно было получить их описание в бескоординатной форме. Его теорема имеет следующий вид:

   $\forall_{abc}(c = a + b \rightarrow a + b = 0 \leftrightarrow c = 0)$

   Прием применяется к условию задачи на описание, имеющей цели "замещение" и "редакция". Вспомогательные задачи такого типа используются решателем при определении геометрических мест точек методами аналитической геометрии. Левая часть преобразуемого равенства имеет сомножителем некоторого своего слагаемого сумму либо натуральную степень суммы. Эта сумма пересекается по своим параметрам с координатами некоторой точки, указанными в посылках задачи. Уровень срабатывания приема равен 3.

Общая стандартизация утверждений

Для стандартизации равенств, содержащих умножение, созданы следующие приемы:

1. Получение явного выражения для известного параметра из линейного соотношения.

   $$\forall_{abc}\left(\neg(a = 0) \rightarrow ab + c = 0 \leftrightarrow b = -\frac{c}{a}\right)$$

   Прием применяется к условию задачи на описание, не содержащему неизвестных. Преобразуемое равенство - либо само условие, либо дизъюнктивный член условия, либо конъюнктивный член дизъюнктивного члена условия. b идентифицируется с переменной, не входящей в a, c. При a равном ± 1 уровень срабатывания равен 0, иначе он равен 1. Таким образом устанавливается приоритет при выборе переменной b, если этот выбор неоднозначен. Прием блокируется при выводе теорем и в задачах с целью "замещение". Последняя цель означает, что ответ должен быть переформулирован без участия неизвестных, с использованием содержащих неизвестные посылок. Очевидно, что в таких задачах разрешение условий относительно известных параметров не нужно.

2. Равенство произведения нулю.

   $\forall_{ab}(ab = 0 \leftrightarrow a = 0 \vee b = 0)$

   Если равенство расположено в условии задачи на описание, то либо оно находится под отрицанием, а само это отрицание - антецедент кванторной импликации, либо задача не имеет цели "свертка". В последнем случае либо содержащее равенство условие имеет неизвестные, либо задача не имеет цели "соединение".

Цель "свертка" возникает крайне редко, в основном при выводе теорем. Она ориентирует на компактную переформулировку условий задачи. Цель "соединение" еще более редка - она задает тенденцию к компактной переформулировке фрагментов ответа, не содержащих неизвестные. Таким образом, рассматриваемый прием применяется почти всегда. Если равенство находится под отрицанием, представляющим собой посылку задачи, либо является условием задачи на описание, имеющей цель "или", то уровень срабатывания равен 0. Иначе он равен 2. Указатель "сопровождение" разрешает выполнение преобразования даже для равенств, участвующих в сопровождении по о.д.з. Утверждения $a = 0, b = 0$ обрабатываются нормализатором "нормчисло", в результате чего могут обратиться в константу "ложь". Тогда заменяющее утверждение будет иметь вид равенства. Указатель "новоеусловие(условие(и(тип(описать)посылка или(заголовок(фикс(0 2 1)ложь)заголовок(фикс(0 2 2)ложь))))x3 заголовок(x3 или)2)" в этих ситуациях понижает до 2 веса всех дизъюнктивных условий задачи на описание, чтобы попытаться усмотреть нереализуемость одной из их альтернатив.

Два дополнительных приема отбрасывают в условии равенства произведения нулю заведомо ненулевые сомножители:

$\forall_{ab}(0 < a \to ab = 0 \leftrightarrow b = 0)$

$\forall_{ab}(\neg(a = 0) \to ab = 0 \leftrightarrow b = 0)$

В действительности эти действия будет выполнять и первый прием: нормализатор "нормлог" преобразует дизъюнкцию с константой "ложь" в остаточное равенство нулю. Поэтому два последних приема нужны лишь для ускорения преобразования в специальных ситуациях. Уровень срабатывания первого из них равен 0; применяется он к посылкам задач на исследование, имеющих цель "известно", причем выражение a должно содержать неизвестные. Уровень срабатывания второго равен 1, и применяется он к равенствам, расположенным внутри квантора. Антецедент первого приема обрабатывается проверочным оператором, второго - непосредственно идентифицируется в контексте.

3. Решение линейного уравнения относительно параметра, связанного квантором.

$$\forall_{abx}\left(\neg(a = 0) \to ax + b = 0 \leftrightarrow x = -\frac{b}{a}\right)$$

Равенство расположено под квантором существования либо является антецедентом кванторной импликации; переменная x входит в связывающую приставку квантора и не входит в a, b. Прием применяется на уровне 6, причем равенство располагается в условии задачи на преобразование. На той же теореме основан еще один прием, применяемый к равенству - условию задачи на описание, имеющей цель "наборантецедентов". Такие задачи используются при выводе теорем для упрощения группы антецедентов полученной новой теоремы. Здесь уровень срабатывания равен 2.

4. Решение линейного уравнения относительно параметра, по которому ведется суммирование либо перемножение.

Теорема приема та же, что в предыдущем пункте. Равенство расположено под описателем "отображение", в связывающую приставку которого входит x. К этому описателю применяется операция "суммавсех" либо "произведениевсех". Уровень срабатывания равен 3.

5. Решение линейного уравнения относительно известного параметра.

   Линейное уравнение с константными коэффициентами a, b, c, являющееся посылкой задачи на исследование, разрешается относительно известного параметра x:

   $$\forall_{abcx}\left(\neg(b=0)\rightarrow a+bx=c\leftrightarrow x=\frac{c-a}{b}\right)$$

   Уровень срабатывания равен 2. Фактически прием был востребован лишь в задачах с целью "линия", при анализе свойств линий, заданных своими уравнениями.

6. Подобные члены по разные стороны числового равенства.

   Приемы относятся к тем сравнительно редким случаям, когда не работает группировка, переносящая подобные члены по одну сторону от равенства:

   $\forall_{abx}(\neg(a-b=0)\rightarrow ax=bx\leftrightarrow x=0)$

   $\forall_{abc}(\neg(a=0)\rightarrow ab=ac\leftrightarrow b=c)$

   В первом случае a, b - константы, причем равенство расположено в посылке задачи на доказательство либо исследование. Уровень срабатывания равен 1. Второй прием применяется без ограничений, на уровне 0. Для ускорения заблокировано использование оператора "алгебрпересечение", т.е. общие множители a определяются по пересечению списков операндов.

7. Сокращение равенства суммы нулю на общий ненулевой множитель всех слагаемых.

   $\forall_{abcd}(a+b=cd\ \&\ \neg(c=0)\rightarrow a+b=0\leftrightarrow d=0)$

   Первый антецедент обращается к нормализатору "факторизация". Прием применяется в задачах на описание, имеющих цель "редуцирование". Такие задачи возникают при выводе теорем. Уровень срабатывания равен 3.

Вывод следствий

В этом подразделе создан единственный прием, усматривающий равенство нулю общего множителя трех равных нулю произведений:

$\forall_{abcd}(ab=0\ \&\ ac=0\ \&\ ad=0\ \&\ \neg(b^2+c^2+d^2=0)\rightarrow a=0)$

Решение уравнения $ax=b$

$$\forall_{abc}\left(a-\text{число}\rightarrow a=bc\leftrightarrow\neg(b=0)\ \&\ c=\frac{a}{b}\ \vee\ b=0\ \&\ a=0\right)$$

Прием применяется к условию задачи на описание. Выражение c содержит неизвестные; a, b - не содержат. Для срабатывания приема необходимо выполнение следующих требований:

1. Если уравнение расположено внутри подтерма условия, то заголовком подтерма служит логическая связка "и", "или" либо квантор существования.

2. Если задача имеет единственную неизвестную, то не существует более простого в смысле эвристической оценки "сложнреш" другого уравнения задачи.

3. Если задача имеет цель "независит", запрещающую вхождение в ответ некоторых переменных, то запрещенные переменные не встречаются в выражении b.

4. Если текущее условие помечено комментарием "серия" и уравнение расположено внутри квантора существования, то оператор "парамописание" должен усматривать целесообразность попытки преобразования данного квантора к виду явного параметрического описания значений неизвестных.

5. Если a - переменная, не входящая в противоположную часть уравнения, то она не связана внешним квантором существования.

Выражение c идентифицируется как произведение всех неизвестных сомножителей. Остаток b должен быть невырожденным. Указатель "новоеусловие(условие(заголовок(x3 синус косинус))x4 контекст(вид(x4 не(равно(x5 0)))контекст(тригаргумент(x5 x6)не(известно(x6))))0)" в случае заголовка "синус" либо "косинус" выражения c понижает до нуля вес всех содержащих неизвестную тригонометрическую операцию отрицаний равенства нулю.

Выражения a, b обрабатываются нормализатором "видумножение". Если этого не сделать немедленно, то после разбора случаев по заменяющей дизъюнкции придется дублировать такие обращения в независимых ветвях решения. Равенство $b = 0$ обрабатывается нормализат орами "нормусм", "нормчисло". Если усматривается отличие b от нуля, то в заменяющий терм вместо $b = 0$ подставляется константа "ложь", а нормализатор "нормлог" превращает дизъюнкцию в равенство $c = a/b$. Таким образом, в большинстве случаев заменяющий терм не имеет заголовка "или". Уровень срабатывания приема равен 1.

При редактировании ответа задачи на описание линейные уравнения с константными коэффициентом и правой частью разрешаются относительно неконстантного известного подвыражения:

$$\forall_{abc}\left(a - \text{число}\ \&\ \neg(b=0) \to ab = c \leftrightarrow a = \frac{c}{b}\right)$$

Здесь b - произведение всех константных множителей левой части; c - константа; a - невырожденное. Равенство расположено в условии задачи на описание, имеющей цель "редакция". Оно не имеет неизвестных. Уровни срабатывания равны 1 либо 5. В первом случае должен существовать комментарий "стандменьше", указывающий, что уже начался этап разрешения сопровождающих по о.д.з. утверждений относительно известных параметров. Во втором случае этот комментарий вводится, инициируя указанный этап.

В случае задач на исследование линейные уравнения в посылках решаются следующими двумя приемами:

$$\forall_{abc}\left(\neg(b=0) \to ab = c \leftrightarrow a = \frac{c}{b}\right)$$

$$\forall_{abc}\left(\neg(c=0) \to ab = c \leftrightarrow a = \frac{c}{b}\ \&\ \neg(b=0)\right)$$

Здесь a идентифицируется с произведением всех неизвестных сомножителей; c известно; b невырождено. Допустимые заголовки надтермов расматриваемого равенства - "и", "или", "существует". Первый прием срабатывает на уровнях 0 либо 1 (0 - для задач на исследование с целью "известно", не имеющих дробных слагаемых правой части); второй - на уровне 2. Он введен для случаев, в которых трудно усмотреть отличие от нуля коэффициента, но легко усматривается отличие от нуля правой части. Первый прием предпринимает попытку разложить на множители b, c, применяя нормализатор "факторизация". Чтобы эта попытка не оказалась чрезмерно громоздкой, введены ограничители трудоемкости. Второй прием имеет ограничитель числа неконстантных дробных степеней в b, c. Для решения геометрических задач на доказательство оказалось удобно выделять некоторые численные переменные как "неизвестные". Это привело к необходимости распространить на их посылки часть аппарат решения уравнений. Оба рассматриваемых приема подпадают под такое распространение: они применимы как в задачах на исследование, так и в задачах на доказательство.

Наконец, упомянем прием, используемый в задачах на исследование для выражения из линейного уравнения одной неизвестной через другие:

$$\forall_{abc}\left(\neg(c=0)\to ab=c\leftrightarrow a=\frac{c}{b}\right)$$

Здесь a - неизвестная, не входящая в b; c известно. Текущая посылка не содержит числовых атомов, отличных от переменных, причем общее число встречающихся в посылках численных неизвестных не менее 4. Выражение b не должно содержать тригонометрических операций с неизвестными. Созданы две версии приема; одна - для задач, в которых выделена прямоугольная система координат, другая - для прочих задач. В первом случае уровень срабатывания равен 7, во втором - 10.

Уравнения $ax = b, cx = d$

Если задача на описание имеет сразу два линейных уравнения такого вида относительно неизвестной x, причем a, b, c, d известны, но не усматривается ни отличие a от нуля, ни отличие c от нуля, а в контексте явно указывается отличие от нуля некоторой линейной комбинации $ae + cf$, то применяется следующий прием, позволяющий обойтись без разбора случаев:

$$\forall_{abcdef}\left(\neg(ae+cf=0)\to ax=b\ \&\ cx=d\leftrightarrow x=\frac{be+df}{ae+cf}\ \&\ (ad-bc)f=0\right)$$

Уровень срабатывания равен 1.

Разложение на множители разности частей уравнения

Это один из наиболее часто используемых приемов решения уравнений элементарной алгебры, особенно в случае тригонометрических операций. Теорема приема имеет следующий вид:

$\forall_{abc}(c=a-b\ \&\ a-\text{число}\ \&\ b-\text{число}\to a=b\leftrightarrow c=0)$

Первый антецедент обращается к нормализатору "видумножение" для получения результата c разложения на множители. Созданы две версии приема; одна срабатывает на уровне 3, другая - на уровне 5. В обоих случаях рассматриваемое уравнение является условием задачи на описание либо посылкой задачи на исследование. При редактировании ответа задачи на описание прием блокируется. Выражение a содержит неизвестные и отлично от переменной. Если b равно 0, то a не имеет своим заголовком один из "мультипликативных" символов "умножение", "степень", "дробь". Если a имеет единственное слагаемое, сомножителем которого служит неизвестный радикал, то проверяется наличие у выражения c неизвестного сомножителя, не имеющего слагаемых такого вида. Введены ограничители трудоемкости; для первой версии приема они обеспечивают сравнительно быструю попытку (не более 3.2 млн. шагов интерпретатора), для второй - более основательную (25 млн. шагов). Первая версия имеет несколько дополнительных фильтров; например, в случае более чем одной неизвестной части уравнения не должны иметь неизвестных дробных слагаемых. Кроме того, в первой версии предусмотрено переключение между обращениями к нормализатору "видумножение" и его ослабленной версией "факторизация". Для этого введены оба нормализатора, сопровождаемые указателями "условие(...)", уточняющими условия применения каждого из них. Если уравнение имеет радикал, под которым расположена сумма, причем число неизвестных более одной, то используется нормализатор "факторизация", иначе - "видумножение". Наконец, обе версии обращаются к пакетному оператору фильтра "фильтрмножителей". Этот оператор анализирует выражение c. Он принимает решение о выполнении преобразования, если выполняется хотя бы одно из следующих требований:

1. Числитель выражения c имеет несколько неизвестных множителей либо имеет своим множителем неизвестную степень.

2. Исходное уравнение имеет тригонометрические операции с различными неизвестными аргументами, а все входящие в c тригонометрические операции с неизвестными имеют равные аргументы.

3. Числитель выражения c имеет своим множителем линейную комбинацию синуса и косинуса неизвестного аргумента, к которой, возможно, добавлены известные слагаемые.

4. Числитель выражения c содержит тригонометрические операции с неизвестными только в виде синуса либо косинуса одного и того же аргумента. Число вхождений таких операций не более 4. Исходное уравнение имеет отличную от синуса и косинуса тригонометрическую операцию с неизвестными.

5. Числитель выражения c имеет единственное вхождение тригонометрической операции с неизвестными, в то время как исходное уравнение имело более одного такого вхождения.

6. Числитель выражения c имеет вид разности степеней с одинаковым натуральным показателем. В нем встречается тригонометрическая операция с неизвестными.

7. Числитель выражения c представим в виде суммы $a + bf(A) + cg(B)$, где f, g - синусы либо косинусы; a, b, c - численные константы, причем $|a| = |b| + |c|$.

8. Числитель выражения c не содержит дробных степеней, в основании которых содержится тригонометрическая операция с неизвестными, а исходное уравнение имеет такие дробные степени. Общее количество вхождений тригонометрических операций с неизвестными в этот числитель не превосходит 2 и меньше, чем у исходного уравнения.

9. Каждая тригонометрическая операция с неизвестными, входящая в c, входит еще в какое-либо уравнение задачи, а для исходного уравнения это неверно. Число неизвестных более одной.

10. Выражение c представимо в виде произведения двух неконстантных множителей, один из которых содержит производную с неизвестными. Текущая задача имеет цель (связка ...). Заметим, что такие цели используются при решении функциональных (фактически - дифференциальных) уравнений для задания списка независимых переменных, варьируемых под неизвестными функциями.

Раскрывание скобок с неизвестными слагаемыми

Если левая часть уравнения имеет неизвестную сумму под знаком произведения либо в основании степени с небольшим натуральным показателем, то может быть предпринята попытка раскрыть скобки. Для этого служат следующие два приема, применяемые в зависимости от того, является ли рассматриваемая сумма сомножителем всей левой части или отдельного ее слагаемого:

$\forall_{abcdfg}(f = a(b+c)^d \to a(b+c)^d = g \leftrightarrow f = g)$

$\forall_{abcdefg}(f = a(b+c)^d + e \to a(b+c)^d + e = g \leftrightarrow f = g)$

Раскрывание скобок выполняется антецедентом, обращающимся к нормализатору "стандплюс". В обоих случаях уровень срабатывания равен 2 либо 3. Показатель степени d идентифицируется с натуральной константой от 1 до 5. Сумма $b+c$ содержит неизвестные; правая часть g известна. Уровень 2 берется, если число неизвестных задачи более одной либо выражение $b+c$ имеет дробное слагаемое; иначе уровень равен 3. Срабатывание приемов оговаривается рядом требований:

1. Если задача имеет единственную неизвестную, то первый прием применяется лишь при условии, что после раскрывания скобок возникает более короткое выражение либо получается квадратное уравнение.

2. Чтобы не усложнить последующих действий по исключению неизвестных радикалов, выражение a не должно иметь своим множителем дробную степень с неизвестным основанием. Однако, если число неизвестных задачи равно 1 и сумма $b+c$ тоже имеет дробную степень с неизвестным основанием, то для первого приема данное ограничение снимается.

3. Первый прием не применяется, если задача имеет целочисленную неизвестную, входящую в рассматриваемое уравнение, причем значение правой части - целое число.

4. Если во втором приеме выражение e имеет дробное слагаемое, то преобразование блокируется: здесь будет применяться другой прием, выполняющий сложение неизвестных дробей.

Для задач на описание, имеющих единственную целочисленную неизвестную, создана версия первого приема, срабатывающая на уровне 5.

Еще два приема раскрывания скобок созданы для устранения неоправданных группировок. На уровне 0 срабатывает следующий прием:

$\forall_{abcdefg}(f = a(b+c) + e \to a(b+c) + e = g \leftrightarrow f = g)$

Он применяется к условию задачи на описание, число неизвестных которой более одной. Выражения a, g известны; сумма $b + c$ не известна и линейна по каждой своей неизвестной. Слагаемое e невырожденное. Прием блокируется, если задача имеет цель (независит ...): наличие группировки могло бы позволить исключить зависимость ответа от переменных выражения a, если сумму $b + c$ сделать тождественно равной 0.

Для устранения группировки, в которой внутренняя сумма $a + b$ имеет неизвестную тригонометрическую операцию, а коэффициент c такой операции не имеет, создан следующий прием:

$\forall_{abcdep}((a + b)c = p \to (a + b)c + d = e \leftrightarrow p + d = e)$

Он применяется к посылке задачи на исследование и срабатывает на уровне 3.

Деление уравнений, левые части которых имеют общий неизвестный множитель

Если левые части двух уравнений задачи на описание имеют общий неизвестный множитель, то одно из них можно упростить, используя прием "деления" уравнений:

$\forall_{abcdef}(\neg(b = 0)\ \&\ ad = b\ \&\ f = be - cd \to ae = c \leftrightarrow f = 0)$

Уравнения являются либо условиями задачи на описание, либо посылками задачи на исследование. Число неизвестных задачи более одной. Выражения a, d, e содержат неизвестные; b, c - не содержат. Второй антецедент идентифицируется со вторым уравнением; третий - определяет левую часть результата "деления" уравнений, которое после устранения дробных выражений сводится к линейной комбинации этих уравнений. Условием эквивалентности перехода является отличие множителя d от нуля, вытекающее из истинности обрабатываемого проверочным оператором первого антецедента. Для срабатывания приема необходимо, чтобы числа неизвестных слагаемых и неизвестных радикалов у нового уравнения были не больше, чем у исходного. Уровень срабатывания равен 3.

На уровне 2 срабатывает упрощенная версия предыдущего приема, в которой результат является линейным уравнением относительно двух неизвестных:

$\forall_{abcdefg}(\neg(fg = 0)\ \&\ ad = bg\ \&\ f = be - cd \to ae = cg \leftrightarrow f = 0)$

Здесь d, e - неизвестные; b, c известны.

Наконец, на уровне 1 во вспомогательных задачах на описание, имеющих цель "стандравно", применяется следующий прием:

$\forall_{abcx}(ax = b \to cx = b \leftrightarrow \neg(x = 0)\ \&\ a = c\ \vee\ x = 0\ \&\ b = 0)$

Задачи с такой целью возникают для решения некоторой подсистемы уравнений, полученной при выводе следствий в задаче на исследование. Выражение x содержит неизвестные; выражения a, b, c никак не ограничены.

Активизация анализатора "неизвмножители"

Кроме пакетного анализатора "сложениеуравнений", генерирующего линейные комбинации уравнений, создан анализатор "неизвмножители", выполняющий деление уравнений, левые части которых имеют общий множитель с неизвестными, а правые части известны. По существу, это частный случай линейных комбинаций, однако анализатор имеет также приемы, подготавливающие возможность деления с сокращением. Приемы анализатора будут рассмотрены ниже, а сейчас укажем лишь прием, активизирующий его:

$\forall_{abc}(a+b=c\rightarrow\emptyset)$

Заголовок приема - "замечание". Консеквент теоремы приема (символ "пусто") - фиктивный, а антецедент идентифицируется с посылкой задачи на исследование, имеющей более одной неизвестной. Левая часть рассматриваемого уравнения должна иметь хотя бы два неизвестных слагаемых; это обеспечивается фильтрами "не(известно(x1))" и "не(известно(x2))". При наличии цели "известно"прием блокируется. Уровень срабатывания равен 2. Обращение к анализатору обеспечивается указателем "анализатор(неизвмножители уровень(2) лимит(3000000))".

Умножение уравнений, имеющих слагаемые, произведение которых известно

Некоторые системы уравнений составляются с использованием специального трюка - перемножения двух выражений с неизвестными, дающего выражение без неизвестных. В качестве такой пары выражений можно взять, например, степени двух взаимно обратных дробей. После того, как в каждом уравнении указанное выражение перенесено в левую часть, а прочие слагаемые - в правую, уравнения перемножаются. Теорема приема, позволяющего решать данные системы, имеет следующий вид:

$\forall_{abcdefg}(g=ad\ \&\ d+e=f\rightarrow a+b=c\leftrightarrow\neg(d=0)\ \&\ g-(f-e)(c-b)=0\ \vee\ d=0\ \&\ a+b=c)$

Второй антецедент идентифицируется с дополнительным уравнением. a, d - перемножаемые выражения. Введен фильтр, проверяющий, что a имеет сомножитель вида $(AX/BY)^m$, а d - сомножитель вида $(CY/DX)^m$ где либо X, либо Y содержит неизвестные. Первый антецедент находит произведение g и упрощает его с помощью вспомогательной задачи. Проверяется, что результат не содержит неизвестных. Для раскрывания скобок в произведении уравнений используется нормализатор "стандплюс". Текущая задача на описание должна иметь более одной неизвестной. Уровень срабатывания приема равен 3.

Решение линейного уравнения относительно переменной связывающей приставки описателя "отображение"

Если отображение задается с помощью условных выражений, причем условие имеет вид равенства, зависящего от варьируемой переменной, то его целесообразно разрешить относительно данной переменной. Для этого создан следующий прием:

$$\forall_{abci}\left(\neg(a=0)\rightarrow ai+b=c\leftrightarrow i=\frac{c-b}{a}\right)$$

Переменная i входит в связывающую приставку внешнего описателя "отображение", причем равенство расположено внутри условного выражения, находящегося под этим описателем. Отсутствует второй описатель "отображение", имеющий общие параметры с рассматриваемым равенством, внутри которого располагался бы первый описатель. Во избежание зацикливания, уточняется приоритет выбора переменной i среди переменных связывающей приставки: берется самая последняя такая переменная. Уровень срабатывания приема равен 0.

Системы уравнений, встречающиеся в задачах на исследование, имеющих цель "известно"

С символом "умножение" связаны следующие приемы преобразования уравнений, используемые в задачах на исследование, имеющих цель "известно":

1. Сокращение на общий ненулевой множитель обеих частей уравнения.

$$\forall_{abcde}\left(\neg(a=0)\to\frac{ab}{d}=\frac{ac}{e}\leftrightarrow\frac{b}{d}=\frac{c}{e}\right)$$

   Прием срабатывает на уровне 1. Знаменатель e невырожденный, а d может обращаться в единицу. Текущая задача - на исследование либо на доказательство. Равенство является ее посылкой. В случае недробных выражений используются следующие приемы:

   $\forall_{abc}(\neg(a=0)\to ab=ac\leftrightarrow b=c)$

   $\forall_{abc}(\neg(a=0)\to ab+ac=0\leftrightarrow b+c=0)$

   Уровень срабатывания равен 1, причем на второй теореме основаны сразу два приема: один из них выделяет общую часть a двух произведений по пересечению групп сомножителей, другой - использует для этого процедуру "алгебрпересечение".

   Следующий прием определяет общую часть a множителей двух слагаемых, после чего обращается к процедуре "факторизация" для суммы e оставшихся слагаемых и усматривает, что она тоже "делится" на a:

$$\forall_{abcdef}\left(\neg(a=0)\ \&\ e=af\to\frac{ab}{c}+ad+e=0\leftrightarrow\frac{b}{c}+d+f=0\right)$$

   Уровень срабатывания равен 3; неконстантная общая часть a определяется по пересечению групп сомножителей.

   Еще одна вариация на ту же тему - сокращение равенства, левая часть которого имеет ровно два слагаемых, а правая ненулевая:

   $\forall_{abcd}(\neg(a=0)\to ab+ac=ad\leftrightarrow b+c=d)$

   В случае планиметрических задач прием блокируется. Уровень срабатывания его равен 4.

   Наконец, сокращение уравнения может происходить на основе дополнительного уравнения, устанавливающего пропорциональность между сомножителями его частей:

   $\forall_{abcdpqn}(\neg(a=0)\ \&\ \neg(b=0)\ \&\ ab=cd\to pa^n=qc^n\leftrightarrow pd^n=qb^n)$

   Третий антецедент представляет собой отдельную посылку задачи; выражения a,c содержат неизвестные, а выражения b,d - не содержат. Прием срабатывает на уровне 7.

2. Выражение одной неизвестной через другую из соотношения пропорциональности.

   $$\forall_{abxy}\left(\neg(a=0)\to ax=by\leftrightarrow x=\frac{by}{a}\right)$$

   Переменные x, y идентифицируются с неизвестными, переменные a, b - с выражениями, не содержащими неизвестных. Уровень срабатывания приема равен 13. Меньшие уровни срабатывания для данного преобразования разрешены в двух специальных случаях. Если задача имеет посылку, представляющую собой уравнение кривой либо поверхности, то создана версия приема, срабатывающая на уровне 5. Наконец, если a - константа, то срабатывание допустимо на уровне 4. Здесь тоже создан отдельный прием, причем он может срабатывать даже при отсутствии цели "известно".

3. Раскрывание скобок.

   Общий случай раскрывания скобок в уравнениях задач на исследование, имеющих цель "известно", реализуется следующим приемом:

   $\forall_{abcefg}(f=a(b+c)+e\to a(b+c)+e=g\leftrightarrow f=g)$

   Выражение $b{+}c$ содержит неизвестные; либо e, либо g отлично от нуля. Уровень срабатывания равен 3. Антецедент раскрывает скобки, используя нормализаторы "стандплюс", "уравнплюс". Прием блокируется, если выполнено хотя бы одно из следующих условий:

   (a) g содержит неизвестные, а левая часть не имеет невырожденных (отличных от переменных и констант) числовых атомов.

   (b) a является невырожденным числовым атомом, а $b+c$ не содержит атомов такого типа.

   (c) Уравнение не имеет невырожденных числовых атомов, причем результат f раскрывания скобок в его левой части не равен правой части.

   (d) Число квадратов сумм в уравнении не менее 4.

   (e) Уравнение имеет хотя бы один из "геометрических" символов "расстояние", "угол", "площадь", "объем" (для таких случаев предусмотрены другие приемы, которые будут приведены в разделах, посвященных элементарной геометрии).

   На той же самой теореме основан еще ряд приемов:

   (a) Приием, применяемый при наличии цели "контроль", активирующей попытки усмотрения нереализуемой ситуации при разборе случаев. Требуется, чтобы существовала другая посылка - уравнение кривой либо поверхности, имеющая общие переменные с рассматриваемой посылкой. Прием блокируется, если a имеет своим множителем дробную степень с неизвестным основанием. Уровень срабатывания равен 3.

   (b) Прием, применяемый к уравнению, имеющему не более двух неизвестных и не содержащему невырожденных числовых атомов. Результат раскрывания скобок f должен быть короче исходной левой части и линеен по всем своим неизвестным. Уровень срабатывания равен 5.

(c) Прием, применяемый, если результат раскрывания скобок f не содержит неизвестных. Левая часть не должна иметь дробных слагаемых; число квадратов сумм в уравнении не более 3. Уровень срабатывания равен 3.

(d) Прием, применяемый, если после раскрывания скобок число неизвестных в левой части уменьшается до одной. Уравнение не имеет невырожденных числовых атомов; число квадратов сумм в нем не более 3. Уровень срабатывания равен 3.

На уровне 2 срабатывает следующий прием, усматривающий, что после раскрывания скобок возникает возможность приведения подобных членов при неизвестном выражении c:

$$\forall_{abcdepqr}(r = a(bc + d) + e(pc + q) \to a(bc + d) + e(pc + q) = r)$$

Здесь выражения a, b, e, p не содержат неизвестных. Прием блокируется, если выражение расположено под описателем "класс", где группировка слагаемых управляется не неизвестными, а переменными связывающей приставки.

На уровне 11 применяется следующий прием вывода:

$$\forall_{abcdep}(p = a(b + c) + d \ \& \ a(b + c) + d = e \to p = e)$$

Усматривается, что a имеет слагаемое вида kX, где k - известный коэффициент, X - не известно; b - слагаемое вида mY, где m - известно, Y - не известно, причем существует посылка вида $nXY + p = q$; n - известно. Таким образом, прием дает новое уравнение с членом вида rXY. Уравнение не должно содержать невырожденных числовых атомов.

4. Решение линейного уравнения для выражения числового атома через численные параметры.

В разделе собраны приемы, не связанные с геометрическими числовыми атомами ("расстояние", "угол", и т.п.), так как для них имеются отдельные приемы, которые будут рассмотрены в главе, посвященной элементарной геометрии.

$$\forall_{abcx}\left(\neg(b = 0) \to a + bx = c \leftrightarrow x = \frac{c - a}{b}\right)$$

x - неконстантный числовой атом, не являющийся переменной и не имеющий своим заголовком символы "расстояние", "угол", "площадь", "длина", "ориентация", "периметр". Равенство не расположено внутри квантора либо описателя, связывающего какие-либо его переменные. Каждый из параметров, входящих в a, b, c, принимает численные значения. Так как утверждение "число(c)" может усматриваться лишь с использованием преобразуемого равенства, то если само c является переменной, проверка этого утверждения не предпринимается. Уровень срабатывания равен 2.

Следующие два приема относятся к случаю, когда x содержит числовой атом, но само с ним не совпадает:

$$\forall_{bcx}\left(\neg(c = 0) \to bx = c \leftrightarrow x = \frac{c}{b} \ \& \ \neg(b = 0)\right)$$

$$\forall_{bcx}(\left(\neg(b=0) \to bx = c \leftrightarrow x = \frac{c}{b}\right)$$

В первом случае отличие коэффициента b от нуля усматривается из отличия от нуля правой части; во втором - усматривается непосредственно из утверждений контекста. Оба приема имеют уровень срабатывания 2. Фильтры их аналогичны фильтрам первого приема.

5. Выражение одного числового атома через другой из соотношения пропорциональности.

$$\forall_{abxy}\left(\neg(a=0) \to ax + by = 0 \leftrightarrow x = -\frac{by}{a}\right)$$

$$\forall_{abxy}\left(\neg(a=0) \to ax = by \leftrightarrow x = \frac{by}{a}\right)$$

На каждой из этих теорем основаны два приема, отличающиеся друг от друга способом идентификации члена by: в одном случае этот член идентифицируется с отдельным слагаемым, в другом - с результатом группировки нескольких слагаемых, имеющих общий числовой атом y. Во всех случаях x, y - невырожденные числовые атомы; a, b не содержат невырожденных числовых атомов. Отсекаются случаи геометрических задач, если только они не содержат физических терминов. Если идентификация выполняется без группировки, то уровень срабатывания равен 2. При наличии группировки - первый прием срабатывает на уровне 4, а второй - на уровне 5. Для срабатывания второго приема в версии "без группировки" необходимо, чтобы выражения a, b не содержали неизвестных, а выражения x, y - содержали. В версии "с группировкой" второй прием требует, чтобы либо a не содержало неизвестных, либо b - содержало.

Чтобы выделить более приоритетные типы числовых атомов при реализации второго приема, созданы еще две его версии "с группировкой". В каждой из них требуется, чтобы a, b были отличны от единицы. Приоритетными числовыми атомами, идентифицируемыми с x, в них считаются некоторые физически величины (например, величина силы). Аналогичную роль играет следующий прием:

$$\forall_{aAB}\left(\neg(a=0) \to ax = y \leftrightarrow x = \frac{y}{a}\right)$$

Здесь x - числовой атом с заголовком "масса"; y - произвольное выражение, не содержащее символов "сила", "Сила" (означающих, соответственно, вектор конкретного силового воздействия либо суммарный вектор всех таких воздействий на тело). Уровень срабатывания данного приема равен 7.

6. Умножение двух уравнений для сокращения на неизвестную сумму.

$$\forall_{abcdpqr}\left((a+b)c = d \ \& \ \frac{p}{q(a+b)} = r \to \frac{cp}{q} = dr\right)$$

Сумма $a+b$ содержит неизвестные, d - не содержит. Уровень срабатывания равен 7. Дополнительно требуется, чтобы в выражение $a+b$ входил символ "расстояние".

7. Усмотрение из соотношения пропорциональности равенства сомножителей.

   $\forall_{abcd}(ab = cd \ \& \ b - d = a - c \ \& \ \neg(b = 0) \ \& \ \neg(c = 0) \to b = c \ \vee \ a = c)$

   Первый антецедент идентифицируется в контексте; выражение b должно представлять собой сумму. Второй антецедент выделен указателем "идентификатор". Он сравнивает обработанные нормализатором "норпмплюс" разности сомножителей b, d и c, a. Уровень срабатывания равен 8.

8. Выражение одной неизвестной через другие из линейного соотношения.

   $$\forall_{abcdp}\left(\neg(a = 0) \ \& \ \neg(b = 0) \to a(bp + c) = d \leftrightarrow p = \frac{d - ac}{ab}\right)$$

   p - неизвестная задачи на исследование, не являющаяся неизвестной внешней задачи на описание. d не должно являться переменной, отличной от неизвестных внешней задачи на описание. Таким образом задается приоритет на исключение "вспомогательных" неизвестных задачи. p не встречается в выражениях a, b, c, d. Уравнение не имеет невырожденных числовых атомов. В геометрических задачах прием блокируется. Уровень срабатывания равен 3.

   Еще один аналогичный прием применяется в задачах, использующих системы координат. Его теорема имеет следующий вид:

   $$\forall_{abcx}\left(\neg(b = 0) \to a + bx = c \leftrightarrow x = \frac{c - a}{b}\right)$$

   Здесь x - неизвестная; c - не неизвестная. Должна существовать посылка, содержащая символ "коорд". Уравнение не имеет невырожденных числовых атомов. Общее число уравнений, не имеющих невырожденных числовых атомов, должно быть не менее 5. Либо уравнение имеет простейший вид $-x = c$, либо число численных неизвестных задачи не менее 4. Переменная x не входит в a, b, c. Уравнение не имеет неизвестных тригонометрических выражений. Если коэффициент b равен ± 1, то уровень срабатывания равен 9, иначе - 11.

9. Деление уравнений для сокращения на множитель, содержащий невырожденные числовые атомы.

   $\forall_{abcxyz}(ax = by \ \& \ \neg(b = 0) \ \& \ \neg(a = 0) \to cx = bz \leftrightarrow cy = az)$

   Первый антецедент идентифицируется с дополнительным уравнением. Неизвестное выражение x содержит невырожденный (отличный от переменной и константы) числовой атом. Коэффициенты a, b, c известны; выражения y, z - не известны и не содержат невырожденных числовых атомов. В результате применения приема возникает уравнение, не содержащее невырожденных числовых атомов. Уровень срабатывания равен 10.

   На той же самой теореме основан еще один прием; у него выражения b, x, y предполагаются содержащими неизвестные, а a, c, z - не содержащими. Результатом служит уравнение с известной правой частью. При этом левая часть cy заведомо проще, чем аналогичная правая часть by дополнительного уравнения, т.к. c известно, а b - нет. Уровень срабатывания равен 3.

   Чтобы исключить невырожденный числовой атом, быть может, сохранив остальные числовые атомы уравнения, служит следующий прием:

$\forall_{abcdx}(ax = b \ \& \ \neg(a = 0) \ \& \ d - \text{число} \to cx = d \leftrightarrow bc = ad)$

Здесь x имеет невырожденный числовой атом A, не входящий в выражения a, b, c, d; выражения b, d не имеют невырожденных числовых атомов. Результатом служит уравнение без A. Могут добавиться невырожденные числовые атомы дополнительного уравнения, не входившие в исходное. В случае геометрических задач прием блокируется; уровень срабатывания равен 11.

В разделе есть еще два аналогичных приема:

$\forall_{abxpq}(\neg(a = 0) \ \& \ by = aq \to ax = bp \leftrightarrow xy = pq)$

$\forall_{abcdxy}(\neg(c = 0) \ \& \ ax = b \to ac = dy \leftrightarrow dxy = bc)$

В первом приеме выражения x, y не имеют невырожденных неизвестных числовых атомов; выражения p, q известны; выражения a, b имеют невырожденные неизвестные числовые атомы. Результатом является уравнение без невырожденных числовых атомов. Уровень срабатывания равен 7. Во втором приеме выражения x, y не имеют невырожденных числовых атомов; выражения b, c, d известны; выражение a не известно. Уровень срабатывания равен 5.

10. Линейная комбинация уравнений для исключения одной из неизвестных.

    Если левые части двух уравнений пропорциональны и содержат численную неизвестную x, не встречающуюся в их правых частях, то берется такая линейная комбинация уравнений, в которой данная неизвестная исключена:

$$\forall_{abcdepqr}\left(\frac{ab}{q} = c \ \& \ \neg(d = 0) \ \& \ p = cdq - bre \to \frac{ad}{r} = e \leftrightarrow p = 0\right)$$

    Первый антецедент идентифицируется с дополнительным уравнением; третий, выделенный указателем "идентификатор", находит левую часть p линейной комбинации уравнений. Выражение a содержит неизвестную, не входящую в c, e; выражения b, q, d, r не содержат неизвестных. Каждая неизвестная и каждый неизвестный числовой атом, входящие в c, входят также в e. В геометрических задачах прием блокируется. Уровень срабатывания равен 4.

11. Ориентация равенства, выражающего один числовой атом через другие.

    Если соотношение представляет один невырожденный числовой атом как произведение другого невырожденного числового атома на коэффицент, не содержащий невырожденных числовых атомов, то оно ориентируется так, чтобы везде в задаче первый атом оказался выражен через второй:

    $\forall_{abc}(bc = a \leftrightarrow a = bc)$

    Здесь a, b - невырожденные числовые атомы; c - коэффициент. Указатель "коммутативно(фикс(0 1))" определяет идентификацию равенства без перестановки его частей, т.е. преобразуемое уравнение ориентировано "неправильно". Чтобы заблокировать прочие приемы перестановки частей равенства, которые могли бы нарушить правильную ориентацию, преобразованное уравнение сопровождается комментарием "ориентацияравенства". Уровень срабатывания равен 1.

Попытка подбора значений неизвестных, обращающих в ноль коэффициент при выражении с варьируемой переменной для устранения зависимости ответа от этой переменной

При нахождении частного решения функционального уравнения (например, дифференциального) приходится подбирать значения численных параметров в выражении, определяющем вид частного решения. Эти параметры играют роль неизвестных задачи на описание, причем независимая переменная x выделяется целью (независит x), блокирующей зависимость от x подбираемых параметров. Чтобы исключить такую зависимость, часто применяется группировка всех членов с x и обнуление не зависящего от x множителя в полученном произведении. Прием, выполняющий данные действия, основан на следующей теореме:

$\forall_{abcdef}(a-b=c \;\&\; c=de+f \;\&\; a - \text{число} \rightarrow a=b \leftrightarrow e=0 \;\&\; f=0)$

Он применяется в задачах на описание, имеющих цель "пример". Первый антецедент группирует все члены уравнения $a=b$ в левой части, обрабатывая их нормализатором раскрывания скобок "стандплюс". В результате возникает выражение c. Указатель "контекст(ключ(цели независит x7)вид(x3 плюс(умножение(x4 x8)x9)) перечень(x4 пересекаются(x7 x4))единица(1 x8)заменазнака(минус x8)единица(0 x9))" идентифицирует с переменной x7 цель (независит ...) и выделяет в c некоторое слагаемое, имеющее пересекающиеся с x7 переменные. Выражение d идентифицируется с произведением всех множителей данного слагаемого, пересекающихся с x7. Второй антецедент выделяет в выражении c все слагаемые, делящиеся на d, и группирует их в выражение de. Это определяется указателем "коэффициент(x5 фикс(2 2 1))". Заменяющее утверждение предпринимает попытку обнулить коэффициент e. Указатель "попытказамены" означает, что вместо необратимого преобразования текущей задачи будет введена вспомогательная задача, в которой и реализуется замена уравнения. Уровень срабатывания равен 4.

Усмотрение существования значения неизвестной, различающего два линейных выражения

$\forall_{abcd}(\exists_x(x - \text{число} \;\&\; \neg(ax+b=cx+d)) \leftrightarrow \neg(a=c) \;\vee\; \neg(b=d))$

Заголовок приема - "связка". Он применяется к задаче на описание, имеющей несущественную неизвестную x. Все содержащие x условия этой задачи суть "$x -$ число, $\neg(ax+b=cx+d)$". Уровень срабатывания равен 1.

Произведение значений неизвестной и известной функций под описателем "отображение"

Прием выносит из-под описателя известный функциональный множитель:

$\forall_{fghP}(\lambda_x(f(x)g(x),P(x))=h \leftrightarrow \forall_x(P(x) \;\&\; g(x)=0 \rightarrow h(x)=0) \;\&\; \forall_x(P(x) \;\&\; \neg(g(x)=0) \rightarrow f(x)=h(x)/g(x)))$

Он применяется к условию задачи на описание. f - неизвестная задачи; выражение h известно. Переменные g, P выделены указателем "отображение", т.е. $g(x), P(x)$ идентифицируются с произвольными термами. Требуется, чтобы выражение $g(x)$ не содержало неизвестных. Указатель "кортежпеременных(x23)" разрешает идентифицировать x с набором из нескольких переменных. Уравнение заменяется на конъюнкцию двух кванторных импликаций. Первая констатирует, что в тех точках, где

$g(x)$ обращается в 0, выражение $h(x)$ тоже равно 0. Вторая - в случае ненулевого $g(x)$ определяет значение $f(x)$ неизвестной функции f. Уровень срабатывания равен 3.

Нормализатор общей стандартизации "нормумножение"

Прежде всего, в нормализатор входят уже упомянутые выше простейшие приемы стандартизации произведений: лексикографическое упорядочение сомножителей, умножение на 0 и на 1, вынесение минуса наружу, умножение десятичных чисел. Кроме того, добавлен ряд приемов, связанных с более "сложными" символами, так как в нормализаторах общей стандартизации эти приемы должны инициироваться по корневой операции умножения:

1. Перемножение десятичных оснований степеней, имеющих одинаковые показатели.

   $\forall_{abc}(0 \leq a \ \& \ 0 \leq b \rightarrow a^c b^c = (ab)^c)$

   $\forall_{abc}(c - \mathrm{rational} \ \& \ \neg(\text{знаменатель}(c) - \mathrm{even}) \rightarrow a^c b^c = (ab)^c)$

   В обоих случаях a, b идентифицируются с десятичными константами.

2. Умножение степеней с одинаковыми основаниями.

   $\forall_{abc}(0 \leq a \rightarrow a^b a^c = a^{b+c})$

   $\forall_{abc}(b - \mathrm{rational} \ \& \ c - \mathrm{rational} \ \& \ \neg(\text{знаменатель}(b) - \mathrm{even}) \ \& \ \neg(\text{знаменатель}(c) - \mathrm{even}) \rightarrow a^b a^c = a^{b+c})$

   Приемы блокируются, если десятичное число умножается на свою дробную степень. Для ускорения введен фильтр "постпозиция(фикс(0 1 2)фикс(0 1 1))", отсекающий попытку идентификации с изменением порядка множителей. Если имеется комментарий "нормуравн", то первый прием применяется только для известных показателей степени b, c. Таким образом сохраняется стандартизация показательных уравнений и неравенств, при которой известные слагаемые показателя степени выносятся в отдельный множитель. Для умножения на степень модуля введен еще один прием:

   $\forall_{abc}(\text{числитель}(b) - \mathrm{even} \ \& \ b - \mathrm{rational} \rightarrow a^b |a|^c = |a|^{b+c})$

3. Умножение степеней с основаниями, отличающимися знаком.

   $\forall_{abc}(\neg(\text{знаменатель}(a) - \mathrm{even}) \ \& \ \text{числитель}(c) - \mathrm{even} \ \& \ a - \mathrm{rational} \ \& \ c - \mathrm{rational} \rightarrow b^c(-b)^a = (-b)^{a+c})$

   $\forall_{abc}(\neg(\text{знаменатель}(a) - \mathrm{even}) \ \& \ \neg(\text{знаменатель}(c) - \mathrm{even}) \ \& \ \neg(\text{числитель}(c) - \mathrm{even}) \ \& \ a - \mathrm{rational} \ \& \ c - \mathrm{rational} \rightarrow b^c(-b)^a = -(-b)^{a+c})$

   Последний прием имеет версию для разности в основании степени:

   $\forall_{abcd}(\neg(\text{знаменатель}(c) - \mathrm{even}) \ \& \ \neg(\text{знаменатель}(d) - \mathrm{even}) \ \& \ \neg(\text{числитель}(d) - \mathrm{even}) \ \& \ c - \mathrm{rational} \ \& \ d - \mathrm{rational} \rightarrow (a-b)^c (b-a)^d = -(a-b)^{c+d})$

4. Умножение модулей.

   $\forall_{ab}(|a||b| = |ab|)$

   Прием можно заблокировать, введя при обращении к нормализатору комментарий "(модуль видумножение)".

5. Умножение на дробь.

$$\forall_{abc}\left(c\frac{a}{b}=\frac{ac}{b}\right)$$

Прием можно заблокировать, введя при обращении комментарий "извлечение".

6. Произведение логарифмов.

$\forall_{abc}((\log_a b)^c \log_b a = (\log_a b)^{c-1})$

Переменная c идентифицируется с натуральной константой, отличной от 1. Если показатель степени вырожденный, применяется другой прием:

$\forall_{ab}(\log_a b \log_b a = 1)$

Уровни срабатывания данных приемов равны 2. Для более общего случая перехода к новому основанию создан прием, срабатывающий на уровне 3:

$\forall_{abc}(\log_a b \log_b c = \log_a c)$

Если текущая задача имеет неизвестные, то для срабатывания необходимо, чтобы a, b, c были известны.

7. Произведение тангенса и косинуса либо котангенса и синуса.

$\forall_{abcdef}(a = \min{(b,c)} \;\&\; d = b{-}a \;\&\; e = c{-}a \rightarrow (\mathrm{tg}\, f)^b(\cos f)^c = (\mathrm{tg}\, f)^d(\sin f)^a(\cos f)^e)$

$\forall_{abcdef}(d = \min{(b,c)} \;\&\; e = b{-}d \;\&\; f = c{-}d \rightarrow (\mathrm{ctg}\, a)^b(\sin a)^c = (\mathrm{ctg}\, a)^e(\cos a)^d(\sin a)^f)$

Показатели степени идентифицируются с натуральными константами, возможно, обращающимися в единицу.

8. Произведение гиперболических тангенса и косинуса, либо гиперболических котангенса и синуса.

$\forall_{abcdef}(a = \min{(b,c)} \;\&\; d = b{-}a \;\&\; e = c{-}a \rightarrow (\mathrm{th}\, f)^b(\mathrm{ch}\, f)^c = (\mathrm{th}\, f)^d(\mathrm{sh}\, f)^a(\mathrm{ch}\, f)^e)$

$\forall_{abcdef}(d = \min{(b,c)} \;\&\; e = b{-}d \;\&\; f = c{-}d \rightarrow (\mathrm{cth}\, a)^b(\mathrm{sh}\, a)^c = (\mathrm{cth}\, a)^e(\mathrm{ch}\, a)^d(\mathrm{sh}\, a)^f)$

9. Произведение тангенса и котангенса.

$\forall_{abcdef}(a = \min{(b,c)} \;\&\; d = b - a \;\&\; e = c - a \rightarrow (\mathrm{tg}\, f)^b(\mathrm{ctg}\, f)^c = (\mathrm{tg}\, f)^d(\mathrm{ctg}\, f)^e)$

10. Произведение модуля на сигнум.

$\forall_a(|a|\mathrm{sg}a = a)$

11. Умножение на O.

При вычислении пределов используется арифметика с выражениями вида $O(x)$. Соответствующие приемы могут обращаться к нормализатору "нормумножение", которому переданы два простейших приема стандартизации таких выражений. Прежде всего, это прием умножения на константу:

$\forall_{ab}(aO(b) = O(b))$

Фильтр "фиксировано(x1 списокпосылок)" проверяет отсутствие посылок вида "стремится(...)" для переменных, входящих в a.

Следующий прием - занесение под O варьируемого множителя, не имеющего вида суммы и не содержащего O:

$\forall_{fgxab}((x \rightarrow a\backslash b) \rightarrow f(x)O(g(x)) = O(f(x)g(x)))$

12. Устранение логарифмов в показателе степени сомножителей.

$$\forall_{abcde}\left(0 < a \ \&\ 0 < b \ \&\ 0 < c \ \&\ \neg(ab - 1 = 0) \to a^{d\log_{ab} c+e} b^{d\log_{ab} c+e} = a^e b^e c^d\right)$$

13. Символы бесконечности.

    $\forall_a(0 < a \to a\infty = \infty)$

    $\forall_a(0 < a \to a(-\infty) = -\infty)$

    Заметим, что в последнем случае бесконечный сомножитель представляет собой символ "минусбеск", а не выражение "минус(плюсбеск)".

14. Умножение константных выражений с радикалами.

    Приведены два несложных упрощающих приема, ориентированных на специальные случаи.

$$\forall_{abcdemn}\left(\left(a + \frac{b\sqrt{c}}{m}\right)\left(d + \frac{e\sqrt{c}}{n}\right) = ad + \frac{bec}{mn} + \frac{bd\sqrt{c}}{m} + \frac{ae\sqrt{c}}{n}\right)$$

    Оба множителя - константные. Либо одно из слагаемых a, d равно 0, либо каждое выражение a, b, c, d, m, n идентифицируется с числом или числовой дробью.

$$\forall_{abc}\left(\sqrt[4]{c}\sqrt{a + b\sqrt{c}} = \sqrt{a\sqrt{c} + bc}\right)$$

    Здесь a, b, c - десятичные константы.

Нормализатор сокращенной записи "упрощумножение"

Этот нормализатор применяется для компактной переформулировки выражений с заголовком "умножение". Кроме приемов устранения вложенных умножений и лексикографического упорядочения операндов, он имеет следующие приемы:

1. Группировка под общий показатель степени.

    Сначала рассматриваются случаи, не требующие проверки неотрицательности оснований обеих степеней:

    $\forall_{abcde}(\neg(\text{знаменатель}(d) - \text{even})\ \&\ \text{числитель}(e) - \text{even}\ \&\ d - \text{rational}\ \&\ e - \text{rational}\ \&\ 0 \leq a \to a^{be}c^{de} = (a^b c^d)^e)$

    $\forall_{abcde}(\neg(\text{знаменатель}(b) - \text{even})\ \&\ \neg(\text{знаменатель}(d) - \text{even})\ \&\ \neg(\text{знаменатель}(e) - \text{even})\ \&\ b - \text{rational}\ \&\ d - \text{rational}\ \&\ e - \text{rational} \to a^{be}c^{de} = (a^b c^d)^e)$

    $\forall_{abc}(a^c b^c = (ab)^c)$

    Для показателей общего вида предусмотрен прием, проверяющий неотрицательность оснований:

$$\forall_{abcdfghj}\left(0 \leq a \ \&\ 0 \leq d \to a^{\frac{cj}{fg}} d^{\frac{bj}{fh}} = \left(a^{\frac{c}{g}} d^{\frac{b}{h}}\right)^{\frac{j}{f}}\right)$$

    Параметры b, c, f, g, h, j могут принимать значение 1, однако либо f либо j единице не равно.

2. Усмотрение разности квадратов.

   $\forall_{abc}((a-b)^c(a+b)^c=(a^2-b^2)^c)$

   $\forall_{abc}((a-b)^c(-(a+b))^c=(b^2-a^2)^c)$

   Здесь выражения a,b не имеют заголовка "плюс". Если a может иметь несколько слагаемых, а b - нет, то применяется следующий прием:

   $\forall_{abcd}(d=a^2-b^2\ \&\ c-\text{rational}\ \&\ \neg(\text{знаменатель}(c)-\text{even})\rightarrow(a-b)^c(a+b)^c=d^c)$

   Первый антецедент обрабатывает разность квадратов нормализаторами "нормплюс", "стандплюс", "упрощплюс". Проверяется, что заменяющий терм короче заменяемого. Фильтр "пересекаются(параметры(x1)параметры(x2))" - ускоряющий: если выражения a,b не имеют общих переменных, то упрощение становится маловероятным. Еще один прием предназначен для выражений специального вида:

   $\forall_{abcpq}(p-\text{rational}\ \&\ \neg(\text{знаменатель}(p)-\text{even})\ \&\ ab=c^q\rightarrow(c^q-a^2-b^2)^p(c^q+a^2+b^2)^p=(-(a^4+b^4+a^2b^2))^p)$

   Третий антецедент сравнивает выражения ab и c^q, применяя к первому нормализатор "нормумножение", а ко второму - "нормстепень". Заметим, что вместо c^q в теореме можно было бы использовать c. Однако, прием нужен в ситуации, где c^q является квадратным корнем из произведения, и применение нормализатора "нормстепень" необходимо для получения из него произведения квадратных корней, т.е. ab. Чтобы оправдать обращение к этому нормализатору, и была выбрана указанная "степенная" версия слагаемого c^q.

3. Усмотрение разности кубов.

   $\forall_{abc}(c-\text{rational}\ \&\ \neg(\text{знаменатель}(c)-\text{even})\rightarrow(a-b)^c(a^2+ab+b^2)^c=(a^3-b^3)^c)$

   Уровень срабатывания равен 1. На уровне 2 применяется небольшое обобщение данного приема:

$$\forall_{abcmn}\left(\left(a^{\frac{m}{n}}-b^{\frac{m}{n}}\right)^c\left(a^{\frac{2m}{n}}+(ab)^{\frac{m}{n}}+b^{\frac{2m}{n}}\right)^c=\left(a^{\frac{3m}{n}}-b^{\frac{3m}{n}}\right)\right)$$

4. Усмотрение суммы кубов.

   $\forall_{abc}(c-\text{rational}\ \&\ \neg(\text{знаменатель}(c)-\text{even})\rightarrow(a+b)^c(a^2-ab+b^2)^c=(a^3+b^3)^c)$

   $\forall_{abc}(c-\text{rational}\ \&\ \neg(\text{знаменатель}(c)-\text{even})\rightarrow(a+b)^c(-a^2+ab-b^2)^c=-(a^3+b^3)^c)$

   Аналогично случаю разности кубов, отдельно рассмотрены дробные показатели степени:

$$\forall_{abcmn}\left(\left(a^{\frac{m}{n}}+b^{\frac{m}{n}}\right)^c\left(a^{\frac{2m}{n}}-(ab)^{\frac{m}{n}}+b^{\frac{2m}{n}}\right)^c=\left(a^{\frac{3m}{n}}+b^{\frac{3m}{n}}\right)\right)$$

5. Вынесение минуса за знак произведения.

   $\forall_{ab}((-a)b=-ab)$

   Прием нужен в тех случаях, когда минус перед сомножителем возник после срабатывания других приемов нормализатора. Заголовком преобразуемого выражения становится "минус", блокирующий дальнейшую обработку произведения. Чтобы избавиться от этой блокировки, введен вспомогательный прием $\forall_{ab}(-ab=-ab)$, реализующий рекурсивное обращение к "упрощумножение" для преобразования ab.

6. Усмотрение суммы либо разности кубов синуса и косинуса.

   В этом случае предварительные преобразования могут несколько видоизменить вид выражений, к которым нужно применить формулу суммы либо разности кубов:

   $\forall_{abc}(c-2a=0\to(\sin a+\cos a)(2b-b\sin c)=2b((\sin a)^3+(\cos a)^3))$

   $\forall_{abc}(c-2a=0\to\sin(a+\pi/4)(2b-b\sin c)=\sqrt{2}b((\sin a)^3+(\cos a)^3))$

   $\forall_{abc}(c-2a=0\to(b\sin a-b\cos a)(2+\sin c)=2b((\cos a)^3-(\sin a)^3))$

   $\forall_{ab}(b-2a=0\to\cos(a+\pi/4)(2+\sin b)=\sqrt{2}((\cos a)^3-(\sin a)^3))$

   Уровень срабатывания равен 3. Использование коэффициента b в первых трех приемах объясняется потребностью идентификации с варьированием знака.

7. Усмотрение синуса двойного угла.

   $\forall_a(\sin a\cos a=\sin(2a)/2)$

8. Умножение на дробь.

   $$\forall_{abc}\left(a\frac{b}{c}=\frac{ab}{c}\right)$$

   Числитель правой части обрабатывается нормализатором "упрощумножение", а вся дробь - нормализатором "упрощдробь".

9. Раскрывание скобок с сокращением.

   $$\forall_{abc}\left(a\left(\frac{b}{a}+c\right)=b+ac\right)$$

   Заменяющий терм обрабатывается с помощью нормализаторов сокращенной записи "упрощплюс", "упрощумножение".

10. Модуль произведения.

    $\forall_{abc}(c=ab\to|ab|=|c|)$

    Прием реализует рекурсивное обращение к обработке операнда модуля, если таковым операндом является произведение. Проверяется, что результат обработки c отличается от исходного выражения.

11. Раскрывание скобок при умножении на степень минус единицы.

    $\forall_{abcd}((-1)^a((-1)^{b-a}c+d)=(-1)^bc+(-1)^ad)$

12. Раскрывание скобок при умножении двух одинаковых радикалов.

    $\forall_{abc}(0\le a\to\sqrt{a}(b\sqrt{a}+c)=ab+c\sqrt{a})$

13. Усмотрение разности либо суммы пятых степеней.

    $\forall_{abc}((a-b)^c(a^4+b^4+a^3b+ab^3+a^2b^2)^c=(a^5-b^5)^c)$

    $\forall_{abc}((a+b)^c(a^4+b^4-a^3b-ab^3+a^2b^2)^c=(a^5+b^5)^c)$

14. Извлечение квадратного корня из десятичного числа.

    $\forall_{ab}(a = b^2 \to \sqrt{a} = b)$

    Переменная a идентифицируется с десятичной константой. Антецедент выделен указателем "программа".

15. Произведение степеней, основания которых отличаются знаком.

    $\forall_{abmn}(m - \text{rational}\ \&\ \neg(\text{знаменатель}(m) - \text{even}) \to (a-b)^n(b-a)^m = -(a-b)^{m+n})$

16. Внесение минуса в сомножитель, имеющий вид разности.

    $\forall_{abc}(-a(b-c) = a(c-b))$

17. Умножение константных сумм с радикалами.

    $\forall_{abcde}((a\sqrt{b}+c)(d\sqrt{b}+e) = ce + adb + (cd+ae)\sqrt{b})$

    Все переменные идентифицируются с десятичными константами.

Нормализатор разложения на множители "видумножение"

Этот нормализатор насчитывает свыше 300 приемов и представляет собой крупнейший пакетный оператор элементарной алгебры. Он предназначен для приведения выражений к "мультипликативному" виду - произведения, степени либо дроби, возможно, с внешним знаком минус. Фактически он объединяет две функции - разложение суммы на множители и сложение дробных выражений, причем в последнем случае реализуются рекурсивные обращения для разложения на множители числителя и знаменателя. Название нормализатора осталось от серии экспериментов с автоматическим синтезом простейших приемов, включая создание новых пакетных операторов и даже автоматический выбор их названия. У операторов, предназначенных для преобразования выражений к заданным заголовкам, название пакета получалось добавлением приставки "вид" к соответствующему заголовочному логическому символу. Если несколько различных заголовков позволяли декомпозировать одно и то же простейшее утверждение, то для них создавался общий пакетный нормализатор. В нашем случае роль такого утверждения играет равенство выражения нулю, а соответствующие заголовки - символы "умножение", "степень", "дробь". Заметим, что до указанной серии экспериментов имелись два различных нормализатора - "разложитьнамножители" и "сложитьдроби". Последующая работа с решателем подтвердила целесообразность их объединения.

Нормализатор имеет 6 уровней срабатывания. Он является корневым, т.е. его приемы применяются лишь к корню преобразуемого выражения. Перейдем к перечислению этих приемов:

1. Вынесение за скобку общего множителя всех слагаемых.

   $\forall_{abc}(ab + ac = a(b+c))$

   Указатель "набор(воройтерм)" разрешает применение приема в случае любого числа слагаемых, большего 1. Общий множитель a всех слагаемых определяется с помощью процедуры "алгебрпересечение". Указатель "единица(1 x2 x3)" означает, что результат сокращения слагаемого на общий множитель может равняться единице. Отсутствие такого указателя для a гарантирует отличие от единицы самого общего множителя. Если при обращении к нормализатору вводится комментарий "константа", то выражение a должно быть неконстантным. Если вводится комментарий "простыеделители", то должно усматриваться, что выражение a целочисленное. Наконец, комментарий "числоценка" вообще блокирует применение данного приема. Этот комментарий вводится, если представляет интерес лишь "невырожденное" разложение на множители. Тогда разрешается вынесение за скобку лишь наибольшего общего делителя r двух натуральных коэффициентов a, c, реализуемое следующим приемом: $\forall_{abcdpqr}(a = pr \ \& \ c = qr \to ab + cd = r(pb + qd))$. Указатель "модификатор" запрещает наличие других слагаемых суммы.

2. Вынесение за скобку общего минуса всех слагаемых.

   $\forall_{ab}(-a - b = -(a + b)$

   Указатель "набор(второйтерм)" разрешает любое число слагаемых. Все они имеют своим заголовком символ "минус".

3. Тождества для непосредственного разложения на множители. Эти тождества обеспечивают быстрое разложение на множители в простейших стандартных ситуациях:

   (a) Произведение двух сумм по два слагаемых.
   $\forall_{abcd}(ac + ad + bc + bd = (a + b)(c + d))$

   Указатель "модификатор" обеспечивает идентификацию при отсутствии других слагаемых. Фильтр "постпозиция(фикс(0 1 2)фикс(0 1 1))" отбрасывает избыточные случаи перестановки первого и второго слагаемых - ac всегда можно считать расположенным в анализируемой сумме перед ad, поменяв при необходимости роли переменных. Указатель "единица(1 x2 x3 x4)" разрешает вырожденные единичные значения переменных b, c, d. Указатель "пересечениесписков(...)" помогает компилятору распознать ситуацию, в которой заголовки слагаемых могут быть отличны от символа "умножение". Отсутствие указателя "заменазнака(минус ...)" объясняется применением при идентификации оператора "алгебрпересечение", который и без этого учитывает возможные знаки "минус" перед слагаемыми.

   (b) Произведение суммы двух слагаемых на сумму трех слагаемых.
   $\forall_{abcde}(ab + ad + ae + bc + cd + ce = (a + c)(b + d + e))$

   Этот прием существенно более трудоемок, чем предыдущий, и блокируется при наличии комментария "нормуравнение", указывающего на ослабленную версию обращения.

   (c) Произведение суммы трех слагаемых на сумму трех слагаемых.
   $\forall_{abcdef}(ab + ad + ae + bc + cd + ce + bf + df + ef = (a + c + f)(b + d + e)$

   Прием крайне трудоемок, однако попытки применить его будут происходить только для сумм длины 9. Чтобы отсечь случаи многочленов, введен фильтр "не(длинатекста(списокпеременных(корень)1))".

   (d) Квадрат суммы.
   $\forall_{bc}((b + c)^2 = b^2 + 2bc + c^2)$

   Замена выполняется справа налево. Указатель "знаксуммы(минус фикс(0 2))" обеспечивает идентификацию в случае, когда перед каждым слагаемым стоит знак "минус". Тогда меняется знак и у заменяющего терма.

Тождество не требует обобщения путем домножения каждого слагаемого на некоторый коэффициент, так как этот коэффициент будет вынесен за скобку другим приемом.

(e) Квадрат разности.

$\forall_{ab}(a^2 - 2ab + b^2 = (a-b)^2)$

Как и в предыдущем приеме, используется указатель "знаксуммы(минус фикс(0 1))", обеспечивающий идентификацию, если все знаки слагаемых изменены на обратные.

(f) Куб суммы.

$\forall_{bc}((b+c)^3 = b^3 + 3bc^2 + 3b^2c + c^3)$

Замена выполняется справа налево.

(g) Куб разности.

$\forall_{ab}(a^3 + 3ab^2 - 3a^2b - b^3 = (a-b)^3)$

(h) Разность квадратов.

В отличие от предыдущих приемов данного пункта, срабатывавших на уровне 1, данный прием срабатывает на уровне 2:

$\forall_{ab}(a^2 - b^2 = (a+b)(a-b))$

(i) Разность кубов.

$\forall_{ab}(a^3 - b^3 = (a-b)(a^2 + ab + b^2))$

Это преобразование не всегда целесообразно, так как приводит к заметному усложнению выражения. Поэтому возникли фильтры, блокирующие его в ряде специальных ситуаций. После применения приема вводится комментарий (6 $a^2 + ab + b^2$), блокирующий попытку разложения второго сомножителя с помощью корней квадратного трехчлена. Этот комментарий будет передан процедуре "видумножение" при рекурсивном обращении для "доразложения" сомножителя. Уровень срабатывания равен 4.

(j) Сумма кубов.

$\forall_{ab}(a^3 + b^3 = (a+b)(a^2 - ab + b^2))$

К предыдущему добавляются указатель "знаксуммы(минус фикс(0 1))", обеспечивающий идентификацию в случае знаков "минус" перед кубами, а также фильтр "постпозиция(фикс(0 1 2)фикс(0 1 1))", блокирующий попытку идентификации с перестановкой слагаемых. При формальном интегрировании используется усиленная версия приема:

$\forall_{abx}(ax^3 + b = (\sqrt[3]{ax} + \sqrt[3]{b})(\sqrt[3]{a^2}x^2 - \sqrt[3]{a}x\sqrt[3]{b} + \sqrt[3]{b^2}))$

Здесь x - переменная интегрирования, не входящая в a, b.

(k) Квадрат разности и разность.

Стандартный шаблон для разложения на множители обычно дает лучшее быстродействие, чем попытки группировки с использованием основных формул сокращенного умножения. Поэтому в нормализаторе создан расширенный запас таких шаблонов. Первый из них - сочетание формул кваадрата разности и вынесения общего множителя за скобку:

$\forall_{abcd}(ca^2 - 2abc + cb^2 + da - db = (a-b)(d + ac - bc))$

Коэффициенты c, d могут обращаться в единицу. Указатель "пересечение(x3 фикс(0 1 1)фикс(0 1 3))" определяет идентификацию переменной

c из рассмотрения только первого и третьего слагаемых. Для устранения избыточности идентификации введено требование, чтобы первое слагаемое лексикографически предшествовало третьему. Уровень срабатывания равен 3.

(l) Квадрат суммы и сумма.
$\forall_{abcd}(ca^2 + 2abc + cb^2 + ad + bd = (a + b)(d + ac + bc))$
Уровень срабатывания - 3.

(m) Разность квадратов и разность.
$\forall_{abcd}(ac + bd^2 - ba^2 - cd = (a - d)(c - ab - bd))$
Коэффиценты b, c могут обращаться в единицу. Уровень срабатывания равен 1.

(n) Разность квадратов и сумма.
$\forall_{abcd}(ab + ac + db^2 - dc^2 = (a + bd - cd)(b + c))$
Коэффициенты a, d могут обращаться в единицу. Уровень срабатывания равен 2.

(o) Разность кубов и разность.
$\forall_{abcd}(ac + ba^3 - bd^3 - cd = (a - d)(c + abd + ba^2 + bd^2))$
Коэффициенты b, c могут обращаться в единицу. Уровень срабатывания равен 3.

(p) Сумма кубов и сумма.
$\forall_{abcd}(ab + ac + db^3 + dc^3 = (a + db^2 + dc^2 - bcd)(b + c))$
Коэффициенты a, d могут обращаться в единицу; допускается одновременная замена знаков слагаемых. Уровень обращения равен 3.

(q) Разность квадрата и квадрата разности.
$\forall_{abc}(a^2 - b^2 - c^2 + 2bc = (a + b - c)(a + c - b))$
Допускается одновременная замена знаков слагаемых. Уровень срабатывания равен 3.

(r) Разность квадрата и квадрата суммы.
$\forall_{abc}(a^2 - b^2 - c^2 - 2bc = (a + b + c)(a - b - c))$

(s) Квадрат суммы трех слагаемых.
$\forall_{abc}(a^2 + b^2 + c^2 + 2ab + 2ac + 2bc = (a + b + c)^2)$
$\forall_{abc}(a^2 + b^2 + c^2 + 2bc - 2ab - 2ac = (b + c - a)^2)$
Уровень срабатывания равен 1.

(t) Остаточная группировка для сомножителя, имеющего вид суммы.

Если одно из слагаемых имеет своим сомножителем сумму, то она становится шаблоном, позволяющим проверить, нельзя ли сгруппировать остальные слагаемые, вынеся эту сумму из них за скобку. Рассмотрим приемы, созданные для такой проверки.

$\forall_{abcd}(c{-}\mathrm{rational}\ \&\ 0 < \text{числитель}(c)\ \&\ 0 < \text{числитель}(c){-}\text{знаменатель}(c) \rightarrow ab^c + bd = (ab^{c-1} + d)b)$

Идентификация начинается с усмотрения слагаемого, имеющего сомножитель вида b^c, где b - сумма; c - рациональная константа, числитель которой больше знаменателя. Указатель "группировка(x1)" определяет идентификацию переменной a путем группировки всех слагаемых, имеющих сомножитель b^c. Остальные слагаемые $S_1, ..., S_m$ обрабатываются в соответствии с указателем "группировки(x2 фикс(0 1 2))". Переменная b к этому моменту идентифицирована с выражением вида $B_1 + \ldots + B_n$. Просматриваются все S_i, представимые как $B_1 k$ для подходящего k, и проверяется, что оставшиеся S_j суть выражения $B_2 k, \ldots, B_n k$ и что $m = n$. Уровень срабатывания равен 2.

В следующем приеме "ключевая" сумма b не расположена под степенью, а является сомножителем:

$\forall_{abd}(ab + bd = (a + d)b)$

Указатели "группировка", "группировки" - такие же, как в предыдущем случае. Уровень срабатывания равен 1.

Наконец, приведем прием, где сумма расположена под рациональной степенью, у которой числитель меньше знаменателя:

$\forall_{abcd}(c - \text{rational} \ \& \ 0 < \text{числитель}(c) \ \& \ \text{числитель}(c) - \text{знаменатель}(c) < 0 \rightarrow ab^c + bd = b^c(a + b^{1-c}d))$

Уровень срабатывания равен 2.

(u) Разность куба суммы трех выражений и суммы их кубов.

$\forall_{abc}((a + b + c)^3 - a^3 - b^3 - c^3 = 3(a + b)(a + c)(b + c))$

Уровень срабатывания равен 1.

(v) Вынесение за скобку модуля.

$\forall_{abc}(ab + |a|c = |a|(\text{sg}(a)b + c)$

(w) Разность пятых степеней.

$\forall_{bc}(b^5 - c^5 = (b - c)(b^4 + b^3c + b^2c^2 + bc^3 + c^4))$

Этот прием, как и приемы для разности и суммы кубов, приводит к существенному усложнению выражения. Поэтому его срабатывание оговаривается рядом условий. Прием для общего случая разности либо суммы одинаковых нечетных степеней без труда программируется на ГЕНОЛОГе, однако целесообразность его применения в обычных ситуациях проблематична - выражение будет усложняться слишком сильно. При необходимости, его можно было бы создать для совсем специальных целевых установок.

(x) Усмотрение произведения двух показательных двучленов.

$\forall_{abcdepqr}(cda^{r+q}/bea^p = 1 \ \& \ \neg(e = 0) \ \& \ 0 < a \rightarrow ba^p + ca^q + da^r + e = (ca^q + e)(da^r + e)/e)$

Остаточная часть суммы e состоит из единственного слагаемого. Первый антецедент выделен указателем "идентификатор"; его левая часть обрабатывается нормализатором "нормдробь". Выражения p, q, r должны быть неконстантными. Уровень срабатывания равен 3.

4. Действия с дробями.

В этом пункте собраны приемы, используемые для предварительной обрабтки дробных слагаемых преобразуемого выражения и для их "сложения":

(a) Сложение дробных выражений.

Прежде всего, рассмотрен случай равных знаменателей:

$$\forall_{abcde}\left(\frac{ad}{c}+\frac{bd}{c}=\frac{(a+b)d}{c}+e\right)$$

Прием срабатывает на уровне 1. Кроме сложения дробей, он выносит за скобку общий множитель числителей d. Фильтр "постпозиция(фикс(0 1 2)фикс(0 1 1))" сокращает поиск пары дробных слагаемых с одинаковыми знаменателями, оставляя для попыток идентификации со второй дробью только те слагаемые, которые идут после первой дроби.

Следующий прием - приведение подобных дробных членов с численными коэффициентами:

$$\forall_{abcdefg}\left(\neg(d=0)\ \&\ \neg(f=0)\to\frac{ab}{cd}+\frac{ae}{cf}+g=\frac{a(bf+de)}{cdf}+g\right)$$

Переменные b,d,e,f идентифицируются с десятичными константами. Нормализаторы обеспечивают выполнение арифметических действий в заменяющем терме. Уровень срабатывания равен 2.

Еще один прием типа приведения подобных членов - усмотрение ситуации, когда множителем числителя дроби служит сумма, образующая часть оставшихся слагаемых преобразуемого выражения:

$$\forall_{abc}\left(\frac{ab}{c}+a=\frac{a(b+c)}{c}\right)$$

Переменная a идентифицируется с суммой. Уровень срабатывания равен 3. Заметим, что компилятор автоматически преобразует теорему к виду

$$\forall_{abcd}\left(\frac{ab}{c}+a+d=\frac{a(b+c)}{c}+d\right),$$

где d - остаток слагаемых. Поэтому a не обязательно составлено из всех оставшихся слагаемых. Вообще, если заменяемый терм имеет вид суммы, то переменную для остатка слагаемых можно вводить в теорему приема вручную , а можно предоставить сделать это компилятору. Чтобы заблокировать последнее, достаточно ввести указатель "модификатор".

Прием для общего случая сложения двух дробных выражений имеет следующую теорему:

$$\forall_{abcdef}\left(\neg(c=0)\ \&\ \neg(d=0)\ \&\ \neg(f=0)\to\frac{ab}{cd}+\frac{ae}{cf}=\frac{a(bf+de)}{cdf}\right)$$

Указатель "единица(1 x1 x2 x3 x4 x5 x6)" разрешает обращаться в единицу любой из переменных a,b,c,d,e,f. Однако, прием относится только к дробным слагаемым, и чтобы отсечь вырожденные случаи, введены фильтры "или(заголовок(фикс(0 1 1)дробь)и(заголовок(фикс(0 1 1)минус)первыйсимвол(фикс(0 1 1)дробь)))", "или(заголовок(фикс(0 1 2)дробь) и(заголовок(фикс(0 1 2)минус)первыйсимвол(фикс(0 1 2)дробь)))". Чтобы выбирать для сложения две самые короткие дроби суммы, введен фильтр "не(контекст(внешоперанд(x7)список(x8 фикс(0 1 1)фикс(0 1 2))короче(x7 x8)или(заголовок(x7 дробь)и(заголовок(x7 минус)первыйсимвол(x7 дробь)))))". Сумма $bf + de$ последовательно обрабатывается нормализаторами "нормплюс", "стандплюс", видумножение" (раскрываются скобки и затем предпринимается попытка разложить на множители). В зависимости от контекста, некоторые из этих обращений могут быть заблокированы либо снабжены дополнительными комментариями. Уровень срабатывания равен 4.

Если имеется единственное дробное слагаемое, то применяются следующие два приема:

$$\forall_{abcd}\left(\neg(b=0)\to\frac{ad}{b}+cd=\frac{(a+bc)d}{b}\right)$$

$$\forall_{abc}\left(\neg(b=0)\to\frac{a}{b}+c=\frac{a+bc}{c}\right)$$

Оба они срабатывают на уровне 4. Первый относится к случаю, когда слагаемых всего два (он имеет указатель "модификатор"). Здесь предпринимается попытка найти и вынести за скобку общий множитель d. Второй прием - остаточный; у него выражение c не имеет дробных слагаемых. Как и при сложении двух дробных выражений, сумма $a + bc$ последовательно обрабатывается нормализаторами "нормплюс", "стандплюс", "видумножение".

(b) Разложение на множители для числителя.

Прежде, чем применять приведенные выше приемы сложения дробных выражений, целесообразно разложить числители и знаменатели на множители: дробь может сократиться либо обнаружатся общие множители числителей или знаменателей складываемых дробей. Перечислим сначала приемы, выполняющие разложение на множители числителя. Точнее, разложению на множители подлежит сумма - основание степени сомножителя числителя:

$$\forall_{abcdef}\left(f=c\ \&\ 0\le c\to a+\frac{bc^d}{e}=a+\frac{bf^d}{e}\right)$$

Переменная c идентифицируется с суммой. Первый антецедент предпринимает попытку разложить эту сумму на множители, используя нормализатор "видумножение". Чтобы отсечь случаи, обрабатываемые другими приемами, проверяется, что показатель степени d не является числовой дробью с нечетным знаменателем. Если выражение c содержит неизвестные, причем показатель степени d дробный, то проверяется отсутствие у c дробных слагаемых. После успешной попытки разложения на множители проверяется неотрицательность c. Чтобы усмотреть ее из сопровождения по о.д.з., лучше брать исходный вид выражения c. Использование этого исходного вида обеспечивается указателем "копия(фикс(2 2))". Чтобы обеспечить сопровождение по о.д.з. для измененного основания степени f, введен указатель "вывод(меньшеилиравно(0 x6)эквивалентно(2))". Он заносит в список посылок нормализатора утверждение $0 \le f$. Это же утверждение регистрируется в комментарии (коррекцияпосылок ...), играющем роль буфера для последующего перенесения полезных новых посылок во внешнюю задачу. Чтобы в ней можно было отбросить уже ненужное утверждение $0 \leq c$, делается пометка о его эквивалентности утверждению $0 \leq f$. Уровень срабатывания равен 2. Если показатель степени при сумме c - числовая дробь с нечетным знаменателем, то применяется другая версия приема:

$\forall_{abcdef}(f = c \;\&\; d - \mathrm{rational} \;\&\; \neg(\text{знаменатель}(d) - \mathrm{even}) \rightarrow a + \frac{bc^d}{e} = a + \frac{bf^d}{e})$

Прием блокируется, если число слагаемых выражения c больше 13. Кроме того, в случае константного c блокируется переход к радикалу, умноженному на сумму, содержащую радикалы. Созданы две версии данного приема - одна применяется при отсутствии в выражении c тригонометрических операций, а другая - при их наличии. Последняя обращается к нормализатору "видумножение" с комментарием "числитель", блокирующим ряд тригонометрических преобразований. В обоих случаях обращение сопровождается комментарием (внешконтекст 0 v), где v - вхождение выражения c в текущее преобразуемое выражение. Этот комментарий позволяет учитывать внешний контекст при принятии решения о целесообразности преобразований.

(c) Разложение на множители для знаменателя.

Приемы аналогичны приведенным выше:

$$\forall_{abcdef}\left(0 < d \;\&\; f = d \rightarrow a + \frac{b}{cd^e} = a + \frac{b}{cf^e}\right)$$

$\forall_{abcdef}(\neg(d = 0) \;\&\; e - \mathrm{rational} \;\&\; \neg(\text{знаменатель}(e) - \mathrm{even}) \rightarrow a + \frac{b}{cd^e} = a + \frac{b}{cf^e})$

(d) Предварительная стандартизация числителя и знаменателя.

В случае тригонометрических выражений разложение на множители числителей и знаменателей бывает неоднозначным, и для получения "сквозных" множителей, выносимых при сложении дробей за скобку, нужно предпринимать дополнительные преобразования фрагментов одной дроби, учитывающие особенности другой дроби. Такие преобразования делаются на двух этапах - до разложения числителей и знаменателей на множители (уровень 1) и после этого разложения, но до сложения дробей (уровень 3). В первом случае используется вспомогательный нормализатор "предвумножение", во втором - "стандумножение". Соответствующие приемы для преобразования знаменателей имеют одну и ту же теорему:

$$\forall_{abcd}\left(\neg(b = 0) \;\&\; d = b \rightarrow \frac{a}{b} + c = \frac{a}{d} + c\right)$$

Второй антецедент обращается к нормализатору, которому передается комментарий (внешконтекст ...). Знаменатель b имеет своим заголовком произведение либо степень; в него входит тригонометрическая либо гиперболическая операция. Для преобразования числителя создан лишь прием, обращающийся к нормализатору "стандумножение":

$$\forall_{abcd}\left(d = a \rightarrow \frac{a}{b} + c = \frac{d}{b} + c\right)$$

Приведем два примера приемов нормализаторов "предвумножение", "стандумножение". Первый из них связан с преобразованием четной степени синуса либо косинуса. Если знаменатель одной дроби имеет множитель $(\sin x)^2$, а другой - множитель $(1-\cos x)$, то перед сложением дробей полезно в первом знаменателе заменить $(\sin x)^2$ на $(1-\cos x)(1+\cos x)$. Это преобразование выполняется следующим приемом нормализатора "стандумножение":

$$\forall_{abcd}\left(d=\frac{c}{2}\to a(\sin b)^c=a(1-\cos b)^d(1+\cos b)^d\right)$$

Выражение c идентифицируется с четной десятичной константой. Чтобы задать условие на внешний контекст, введен фильтр: "контекст(внешконтекст(x5)соотвмножитель(x5 x6) вид(x6 умножение(x7 степень(x8 x9)))единица(1 x7 x9)или(контекст(вид (x8 плюс(1 степень(косинус(x2)x10)))единица(1 x10) натуральное(x10) не(четное(x10)))контекст(вид(x8 плюс(1 минус(степень (косинус(x2)x10))))единица(1 x10) натуральное(x10)) контекст (вид(x8 плюс(степень(косинус(x2)x10)минус(1)))единица(1 x10) натуральное(x10))))". Идентифицирующий терм "внешеконтекст(x5)" присваивает переменной x5 вхождение преобразуемого нормализатором выражения во внешний контекст (в числитель либо знаменатель дроби). Идентифицирующий терм "соотвмножитель(x5 x6)" перечисляет в качестве значения переменной x6: вхождение противоположного операнда той же дроби, а также вхождения одноименного операнда (числителя; знаменателя) других дробных слагаемых внешней суммы. Далее проверяется наличие сомножителя выражения x6, основание степени которого представимо как $(1\pm(\cos b)^{2k+1}$. Эти условия гаранитруют либо возможность сокращения дроби после преобразования, либо получение общего множителя ее с другой дробью.

В качестве примера приема нормализатора "предвумножение" рассмотрим следующее тригонометрическое преобразование, позволяющее получить из сумм с тангенсами уже встречающийся в контексте косинус двойного аргумента:

$$\forall_{abcd}\left(d=\cos(2a)\to(1+\operatorname{tg}a)(b-b\operatorname{tg}a)c=\frac{bcd}{(\cos a)^2}\right)$$

Фильтр "контекст(внешконтекст(x5)соотвмножитель(x5 x6)вид(x6 умножение(x7 степень(x4 x8)))единица(1 x7 x8))" выявляет вхождение косинуса x4 двойного аргумента либо в противоположном операнде той же дроби, либо в одноименном операнде другой дроби.

(e) Сокращение на общий множитель оснований двух степеней с одинаковыми показателями.

$$\forall_{abcdefg}\left(\neg(b=0)\to\frac{a(bc)^d}{e(bf)^d}+g=\frac{a\left(\frac{c}{f}\right)^d}{e}+g\right)$$

Прием срабатывает на уровне 3. Он нужен в тех случаях, когда степень произведения не удается преобразовать к виду произведения степеней из-за неопределенности знаков сомножителей.

(f) Устранение иррациональности в знаменателе.

Перед сложением дробных выражений обычно предпринимается устранение иррациональности в знаменателях. Уровень срабатывания равен 1. Отдельно рассматриваются случаи неконстантных и константных выражений с радикалами. В первом случае теорема приема такова:

$\forall_{abcdefghpq}(\neg(a+b=0)\ \&\ g=a-b\ \&\ p=a^2\ \&\ q=b^2\ \&\ c=p-q\ \&\ (\neg(c=0)\ \vee\ \neg(g=0))\rightarrow\ f/(d(a+b)^e)+h=f(g/c)^e/d+h)$

Сумма с иррациональностями - $a+b$; она должна быть неконстантной. Выражение a имеет сомножитель, представляющий собой степень с дробным показателем вида $m/2n$. Основание этой степени содержит сумму. Если b имеет более одного слагаемого, то ни одно из них не имеет своим сомножителем степень с дробным показателем и неконстантным основанием либо степень с неконстантным показателем. Третий и четвертый антецеденты возводят a и b в квадрат, используя нормализатор общей стандартизации "нормстепень" и нормализатор раскрывания скобок "стандплюс". После этого пятый антецедент получает результат c разложения на множители разности a^2-b^2. Далее проверяются следующие условия:

i. У c нет слагаемых, имеющих среди своих множителей степени с дробным показателем вида $m'/2n'$;
ii. Число слагаемых у выражения c меньше суммарного числа слагаемых p,q т.е. при вычитании p и q произошли сокращения;
iii. Выражение $a+b$ не имело дробных слагаемых;
iv. Выражение f не имеет сомножителя, имеющего вид суммы с дробным слагаемым;
v. Выражение h не является дробью, знаменатель которой делит d.

Шестой антецедент проверяет корректность преобразования - отличие от нуля выражения $a-b$, на которое (после возведения его в степень e) домножались числитель и знаменатель. Он имеет вид дизъюнкции, так как заранее трудно сказать, что легче проверить - отличие от нуля $a-b$ либо преобразованной разности a^2-b^2. Указатель "вывод(не(равно(x3 0))эквивалентно(1))" сохраняет для дальнейшего сопровождения по о.д.з. посылку $\neg(c=0)$.

Если сумма $a+b$ находится под модулем, используется немного модифицированный прием:

$\forall_{abcdefghpq}(\neg(a+b=0)\ \&\ g=|a-b|\ \&\ p=a^2\ \&\ q=b^2\ \&\ c=|p-q|\ \&\ (\neg(c=0)\ \vee\ \neg(g=0))\rightarrow\ f/(d(|a+b|^e)+h=f(g/c)^e/d+h)$

Для константной суммы с радикалами вместо обращений к нормализаторам общего вида можно использовать непосредственные арифметические вычисления. Здесь созданы следующие два приема:

$$\forall_{abcdefgh}\left(g=c^2-d^2e^f\ \&\ \neg(g=0)\rightarrow\frac{a}{b\left(c+de^{\frac{f}{2}}\right)}+h=\frac{a\left(c-de^{\frac{f}{2}}\right)}{bg}+h\right)$$

$$\forall_{abcdefghpq}\left(p=c^2e^g-d^2f^h\ \&\ \neg(p=0)\rightarrow\frac{a}{b\left(ce^{\frac{g}{2}}+df^{\frac{h}{2}}\right)}+q=\frac{a\left(ce^{\frac{g}{2}}-df^{\frac{h}{2}}\right)}{bp}+q\right)$$

В первом случае переменные c,d,e идентифицируются с десятичными константами; f - с натуральной константой, меньшей 5. Во втором случае

c, d, e, f идентифицируются с десятичными константами; g, h - с натуральными константами, меньшими 5. Антецеденты выделены указателем "программа"; они компилируются в процедуру, реализующую арифметические вычисления и проверяющую отличие результата от нуля.

(g) Преобразование степени дроби к виду частного степеней.

$$\forall_{abcde}\left(e - \text{rational}\ \&\ \neg(\text{знаменатель}(e) - \text{even}) \to a\left(\frac{b}{c}\right)^{e} + d = \frac{ab^e}{c^e} + d\right)$$

Уровень срабатывания равен 3.

(h) Занесение внешнего множителя в числитель.

$$\forall_{abcd}\left(a + b\frac{c}{d} = a + \frac{bc}{d}\right)$$

Уровень срабатывания равен 1.

(i) Преобразование к виду дроби степени с отрицательным показателем.

$$\forall_{abcd}\left(ab^{-c} + d = \frac{a}{b^c} + d\right)$$

Прием используется только для константных выражений. Уровень срабатывания равен 1.

5. Доразложение на множители операндов мультипликативного выражения.

Нормализатор "видумножение" - корневой, т.е. его приемы применяются только к самому преобразуемому выражению, а не к подвыражениям. Некоторое исключение составляют случаи, когда теорема приема явно выделяет лишь часть корневых слагаемых, не изменяя прочих слагаемых. Однако, после того, как выражение оказалось представленным в виде произведения, дроби либо степени, нужно попытаться продолжить разложение на множители сомножителей этого произведения, числителя или знаменателя дроби, основания степени. Здесь используются рекурсивные обращения к нормализатору "видумножение", реализуемые следующими приемами:

(a) Разложение на множители основания степени.

$\forall_{abc}(c - \text{rational}\ \&\ \neg(\text{знаменатель}(c) - \text{even})\ \&\ a = b \to a^c = b^c)$

$\forall_{abc}(0 \leq a\ \&\ a = b \to a^c = b^c)$

Доразложению подлежит выражение a, которое не обязательно имеет своим заголовком символ "плюс". Второй прием применяется, если показатель степени не является числовой дробью с нечетным знаменателем. Антецедент $a = b$ выполняет обращение к нормализатору "видумножение", передавая ему комментарий (контекст v), где v - вхождение преобразуемого основания степени в степенной терм. Этот комментарий может быть полезен при принятии решения о целесообразности тех или иных преобразований a. Второй прием снабжен указателем "вывод(меньшеилиравно(0 x2)эквивалентно(1))", обеспечивающим ввод используемой при сопровождении по о.д.з. посылки $0 \leq b$. Уровень срабатывания этих и перечисляемых ниже приемов "доразложения" равен 1.

   (b) Разложение на множители сомножителя.
   $\forall_{abc}(a = b \to ac = bc)$

   (c) Разложение на множители числителя.
   $\forall_{abc}(a = b \to \frac{a}{c} = \frac{b}{c})$

   (d) Разложение на множители знаменателя.
   $\forall_{abc}(a = b \ \& \ \neg(a = 0) \to \frac{c}{a} = \frac{c}{b})$

   (e) Разложение на множители выражения под минусом.
   $\forall_{ab}(a = b \to -a = -b)$

   Этот прием необходим, чтобы внешний минус не заблокировал доразложение.

6. Доразложение на множители суммы - основания степени сомножителя слагаемого.

   Если сумма находится в основании степени слагаемого, то разложение ее на множители "измельчает" сомножители данного слагаемого. Новые сомножители можно будет вынести за скобку либо использовать при группировках. Поэтому данное преобразование представляет собой предварительную стандартизацию, полезную для дальнейших действий. Оно выполняется следующими приемами, срабатывающими на уровне 2:

   $\forall_{abcdef}(f = a + b \ \& \ 0 \leq a + b \to d(a + b)^c + e = df^c + e)$

   $\forall_{abcdef}(f = a + b \ \& \ 0 < a + b \to d(a + b)^c + e = df^c + e)$

   $\forall_{abcdef}(f = a+b \ \& \ c - \mathrm{rational} \ \& \ \neg(\text{знаменатель}(c)\mathrm{even}) \to d(a+b)^c + e = df^c + e)$

   В первом приеме показатель степени c константный; во втором - неконстантный. Проверка неотрицательности либо положительности основания $a+b$ собственно для преобразования не нужна, однако она позволяет вывести в первом случае следствие $0 \leq f$, а во втором случае - следствие $0 < f$. Они необходимы для дальнейшего сопровождения по о.д.з., так как возникает новая степень f^c. Преобразование блокируется, если выражение $a + b$ расположено под радикалом, содержит неизвестные и имеет дробное слагаемое. Это делает более приоритетными преобразования типа "возведения уравнения в квадрат", исключающие неизвестные радикалы. Первый антецедент обращается к нормализатору "видумножение"; проверяется, что результат либо приобрел мультипликативный заголовок, либо отличен от исходного выражения и является основанием степени, входящей в e. Впрочем, для последнего приема данное требование несколько видоизменено: просто проверяется, что f не имеет заголовка "плюс".

7. Квадратный трехчлен.

   При разложении на множители квадратного трехчлена используются следующие приемы:

   (a) Разложение в общем случае.

$$\forall_{abcde}\left(e^2 = b^2 - 4ac \to ad^2 + bd + c = \frac{(2ad + b - e)(2ad + b + e)}{4a}\right)$$

Используется ряд фильтров, отбрасывающих нецелесообразный выбор выражения d. Оно должно быть неконстантным, причем его основание степени не должно встречаться ни в b, ни в c. Указатели "группировка(x1)", "группировка(x2)" обеспечивают идентификацию a, b путем группировки всех членов, соответственно, вида kd^2 и kd. В этих членах сомножители d^2, d появляются явным образом. Отбрасывается случай, когда при группировке коэффициент b оказался равным нулю: здесь будут работать другие приемы. Следующий ограничитель - требование, чтобы дискриминант после разложения на множители оказывался полным квадратом. Оно накладывается видом антецедента: правая его часть обрабатывается нормализаторами и затем идентифицируется e. Чтобы сразу отсекать случаи, когда данное условие не выполняется, добавлен ускоряющий фильтр, проверяющий, что все параметры выражения b встречаются в a, c. Комментарии (3 ...), (6 ...) блокируют применение приема, если заранее известно, что дискриминант отрицателен (например, для трехчлена $x^2 + ax + a^2$, получающегося по формуле суммы кубов).

Заметим, что о.д.з. заменяющего выражения, если не удается сократить его на неконстантный множитель a, должно включать условие отличия этого a от нуля. В то же время, исходное выражение может рассматриваться для любых a. Чтобы избежать такого рода коллизий, прием снабжен фильтром "контрольодз", проверяющим сохранение о.д.з. Фактически это означает контроль сокращения тех сомножителей a, отличие которых от нуля неочевидно.

Созданы две версии приема; уровни срабатывания их равны 3 и 5. Если выражение d имеет тригонометрическую операцию, параметры которой встречаются в тригонометрических операциях коэффициентов a, b, c, то прием уровня 3 блокируется: делается пауза для возможных срабатываний чисто тригонометрических приемов. Наоборот, прием уровня 5 предназначен только для таких особых случаев.

(b) Усмотрение возможности представить трехчлен в виде разности квадрата суммы и квадрата некоторого выражения.

Иногда сумма, представляющая собой квадратный трехчлен относительно некоторого выражения X, не может быть разложена на множители с помощью предыдущего приема из-за отрицательности дискриминанта, однако разлагается на множители другими средствами. Например, выражение $a^4 + b^4 + a^2b^2$, являющееся квадратным трехчленом относительно a^2, разлагается на множители с помощью искусственного приема - прибавления и вычитания a^2b^2 для получения разности квадратов. Такой прием можно представить следующим образом:

$\forall_{abcd}(d^2 = (2-c)ab \rightarrow a^2 + b^2 + abc = (a+b-d)(a+b_d))$

Выражение c идентифицируется с десятичной константой. Чтобы допустить возможность вырожденного случая $c = 0$, введен указатель "подстановка(фикс(0 1 3)x3 0)". Указатель "пересечениесписков(фикс(0 1 3))" допускает возможность отличия заголовка члена abc от символа "умножение" - если b, c одновременно равны 1. Антецедент преобразует выражение $(2-c)ab$, используя нормализатор "видумножение". Предварительно нормализатор "нормплюс" применяется к сумме $2-c$. Если результат разложения на множители оказался полным квадратом, реализуется указанная в теореме замена. Прием срабатывает на уровне 2.

(c) Усмотрение возможности представить трехчлен в виде разности квадрата разности и квадрата некоторого выражения.

Аналогично предыдущему случаю, создан прием для выделения квадрата разности:

$\forall_{abcd}(d^2 = -(c+2)ab \to a^2 + b^2 + abc = (a-b-d)(a-b+d))$

(d) Разложение на множители коэффициентов квадратного трехчлена, имеющих вид суммы.

Прием предпринимает попытку вынести за скобку общий множитель a коэффициентов трехчлена после разложения их на множители. Иногда такое преобразование существенно упрощает выкладки при доразложении трехчлена.

$\forall_{abcdefgh}(ab = c \,\&\, ad = e \,\&\, af = g \to ch^2 + eh + g = a(bh^2 + dh + f))$

Коэффициенты c, e, g имеют заголовок "плюс"; h - переменная, не входящая в g. Указатели "группировка(x3)", "группировка(x5)" определяют идентификацию переменных c, e путем группировки всех членов с h^2 и с h. Уровень срабатывания равен 2.

(e) Разложение на множители квадратного трехчлена при интегрировании.

В специальных случаях появление радикалов после разложения на множители квадратного трехчлена является допустимым. Для них созданы дополнительные приемы. Первым примером такого рода служит разложение на множители при формальном интегрировании:

$\forall_{abcde}\left(e = b^2 - 4ac \,\&\, 0 \le e \,\&\, f = \sqrt{e} \to ad^2 + bd + c = \frac{(2ad+b-f)(2ad+b+f)}{4a}\right)$

Требуется наличие комментария (нормИнтеграл x), указывающего на переменную интегрирования x, которая должна входить в d и не входить в a, b, c. Если x встречается в d внутри тригонометрической операции, то прием блокируется. Вместо комментария (нормИнтеграл x) может использоваться комментарий (переменная x). Он указывает, что представляют интерес лишь разложения, имеющие не менее двух множителей с переменной x. Уровень срабатывания равен 3.

(f) Разложение на множители квадратного трехчлена при исследовании вида поверхности второго порядка.

$\forall_{abcde}\left(pe^2 = b^2 - 4ac \,\&\, 0 \le p \to ad^2 + bd + c = \frac{(2ad+b-e\sqrt{p})(2ad+b+e\sqrt{p})}{4a}\right)$

Имеется комментарий (переменные $x\ y\ z$), определяющий координатные переменные. Они не должны входить в остаточный после выделения в дискриминанте полного квадрата множитель p. Уровень срабатывания равен 3.

8. Частичные группировки. На достаточно высоких уровнях начинаются попытки сгруппировать несколько слагаемых. Все они имеют рекурсивный характер: прием группирует слагаемые и сразу обращается к нормализатору "видумножение" для попытки разложения данной версии группировки на множители. Это обращение сопровождается комментарием "группировка", блокирующим раскрывание скобок ("разгруппировку"). Попытки группировки выполняются следующими приемами:

(a) Разность кубов.
$\forall_{abcde}(e = a(b-c)(b^2+bc+c^2)+d \to ab^3-ac^3+d=e)$

Антецедент применяет нормализатор "видумножение" к выражению, полученному после группировки и разложения на множители разности кубов. Проверяется, что результат e имеет мультипликативный заголовок. Дополнительно проверяются следующие условия:

i. Если есть комментарий "группировка", то среди слагаемых выражения d имеется хотя бы одно, делящееся на некоторую (быть может, первую) степень $b-c$.
ii. Число слагаемых преобразуемого выражения меньше 9.
iii. Если нет комментария "группировка", то число слагаемых преобразуемого выражения не меньше 4.
iv. Преобразуемое выражение неконстантное.
v. Выражение d не имеет дробных слагаемых.
vi. Если есть комментарий "группировка", то d не является суммой, имеющей константное слагаемое.
vii. Либо b, либо c не является константой.
viii. Ни b, ни c не является степенью с неизвестным показателем.

Созданы две версии приема, отличающиеся ограничениями на преобразование выражений с тригонометрическими операциями. Уровень срабатывания равен 4.

(b) Сумма кубов.
$\forall_{abcde}(e = a(b+c)(b^2-bc+c^2)+d \to ab^3+ac^3+d=e)$

Аналогично предыдущему.

(c) Разность квадратов.
$\forall_{abcde}(e = a(b-c)(b+c)+d \to ab^2-ac^2+d=e)$

Если есть комментарий "группировка", то среди слагаемых выражения d должно иметься хотя бы одно, делящееся на некоторую степень $b-c$ либо $b+c$. Остальное - аналогично предыдущему.

(d) Квадрат суммы.
$\forall_{abcde}(e = a(b+c)^2+d \to ab^2+2abc+ac^2+d=e)$

Число слагаемых преобразуемого выражения равно 4. Это требование означает, что обычно слагаемое d уже является результатом предшествующих группировок. При наличии комментария "группировка" оно должно делиться на степень $b+c$. Еще один прием для квадрата суммы связан с радикалами, так как здесь нужно учитывать возможность сокращений:
$\forall_{abcdefg}(e = ab+ac \ \& \ d=e+f \ \& \ g=a(\sqrt{b}+\sqrt{c})^2+f \to 2a\sqrt{b}\sqrt{c}+d=g)$

Идентификация происходит в следующем порядке: по слагаемому $2a\sqrt{b}\sqrt{c}$ определяются a, b, c. Затем первый антецедент вычисляет e, обращаясь к нормализатору "стандплюс"; второй - усматривает, что e составляет сумму части слагаемых выражения d. При этом доопределяется f. Наконец, третий антецедент предпринимает попытку разложить результат группировки на множители. Проверяется, что число слагаемых выражения e меньше суммы чисел слагаемых выражений b, c, т.е. что имело место сокращение либо приведение подобных членов. Очевидно, в такой ситуации непосредственная идентификация квадрата суммы первым приемом будет затруднена.

(e) Квадрат разности.

$\forall_{abcde}(e = a(b-c)^2 + d \to ab^2 - 2abc + ac^2 + d = e)$

Аналогично предыдущему.

(f) Куб суммы.

$\forall_{abcde}(e = a(b+c)^3 + d \to ab^3 + 3abc^2 + 3acb^2 + ac^3 + d = e)$

Число слагаемых равно 5, в остальном - аналогично двум предыдущим приемам.

(g) Куб разности.

$\forall_{abcde}(a = a(b-c)^3 + d \to ab^3 + 3abc^2 - 3acb^2 - ac^3 + d = e)$

(h) Разность пятых степеней.

$\forall_{abcde}(e = a(b-c)(b^4 + b^3c + b^2c^2 + bc^3 + c^4) + d \to ab^5 - ac^5 + d = e)$

Фильтры аналогичны случаю разности квадратов. Число слагаемых должно быть не более четырех.

(i) Использование уникального множителя.

Если слагаемое преобразуемого выражения имеет такой сомножитель, который не появляется в других слагаемых ("уникальный множитель"), то при некоторых дополнительных допущениях можно сделать эвристическое заключение о невозможности разложения на множители и сразу прервать работу нормализатора:

$\forall_{abc}(ab + c = ab + c)$

Здесь a - сомножитель, не имеющий вида полного квадрата, куба либо пятой степени; основание степени выражения a не встречается среди оснований степени сомножителей слагаемых выражения c и не является десятичной записью числа. Преобразуемая сумма не имеет тригонометрических операций либо дробных слагаемых и не допускает раскрывания скобок: среди сомножителей ее слагаемых не встречаются степени сумм. Тогда работа нормализатора обрывается, и текущее выражение выдается в качестве ответа. Это обеспечивается указателем приема "выход". Уровень срабатывания равен 2.

Другой прием, использующий "уникальный множитель" a, предпринимает группировку всех слагаемых с этим множителем, после чего отдельно разлагает на множители коэффициент при a и сумму остальных слагаемых. Таким образом, задача декомпозируется на две более простых задачи:

$\forall_{abcdef}(de = ab \;\&\; df = c \to ab + c = d(e+f))$

Уникальный множитель a удовлетворяет приводимым ниже ограничениям. Переменная b, выделенная указателем "группировка", идентифицируется путем группировки всех слагаемых с множителем a. Она должна представлять собой сумму либо иметь сомножитель - степень суммы. Первый и второй антецеденты обращаются к нормализатору "видумножение" для разложения на множители выражений b, c. После этого идентифицируется общая часть d найденных разложений. Сумма $e + f$ в заменяющем терме обрабатывается нормализатором "стандплюс", и выдается результат. Как и в предыдущем случае, прием снабжен указателем "выход". Условия на a сводятся к следующему:

   i. Основание степени выражения a не делит основания степени слагаемых выражения c.
   ii. Выражение a не является степенью десятичной константы либо натуральной степенью суммы.
   iii. Если a - дробная степень, то b не содержит дробной степени с тем же основанием.
   iv. Если a имеет вид радикала от суммы, то c не является суммой.

Если отсутствует комментарий "группировка", то слагаемые преобразуемого выражения не имеют своими множителями сумм и натуральных степеней сумм. Преобразуемое выражение не содержит тригонометрических операций и дробных слагаемых. Число его слагаемых не менее 4. Прием срабатывает на уровне 2, отсекая попытки более высоких уровней (в частности, попытки не "уникальных" группировок).

9. Разложение многочленов путем подбора целочисленных коэффициентов.

   Для разложения на множители многочленов с целочисленными коэффициентами созданы несколько приемов, использующих соображения делимости коэффициентов. Они относятся к степеням от 3 до 6, так как далее трудоемкость данного метода сильно возрастает. Одновременно с этим, на ГЕНОЛОГе реализован алгоритм Берлекэмпа, позволяющий находить разложения на множители целочисленных многочленов произвольных степеней. Он используется для многочленов, степени которых более 4. Реализация алгоритма Берлекэмпа будет рассмотрена ниже, здесь же рассмотрим приемы, основанные на делимости:

   (a) Представление многочлена третьей степени в виде произведения линейного множителя и квадратного трехчлена.

      $\forall_{abcdefghijk}(fh = d \;\&\; gj = a \;\&\; 0 < g \;\&\; gi = b - fj \;\&\; gh + fi = c \to ae^3 + be^2k + cek^2 + dk^3 = (fk + ge)(hk^2 + iek + je^2))$

      Коэффициенты a, b, c, d идентифицируются с целочисленными константами. Они могут обращаться в единицу. Коэффициенты b, c могут обращаться в ноль; чтобы обеспечить идентификацию в этом случае, введены указатели "подстановка(фикс(0 1 2)x2 0)", "подстановка(фикс(0 1 3)x3 0)". Фактически прием рассматривает не обычный многочлен третьей степени от одной переменной, а однородный многочлен от двух переменных - e, k. Чтобы разрешить одной из них обращаться в единицу и таким образом включить в область действия приема многочлены от одной переменной, добавлен указатель "пересечениесписков(фикс(0 1 2)фикс(0 1 3))". Он отбрасывает идентификацию заголовка "умножение" у второго и третьего членов. Указатели "перечень(x1 десчисло(x1))", "перечень(x4 десчисло(x4))" определяют идентификацию коэффициентов a, d как произведений всех числовых множителей первого и четвертого членов. Эти члены дают возможность сразу идентифицировать также e, k.

      Указатель "модификатор" запрещает наличие слагаемых, не идентифицированных с указанными в теореме четырьмя членами. Введены ускоряющие фильтры "меньше(количествоoперандов(теквхожд)5)", "меньше(2 количествооперандов(теквхожд))", сразу отсекающие случаи более чем четырех или менее трех слагаемых. Фильтр "постпозиция(фикс(0 1 4)фикс(0 1 1))" отсекает рассмотрение симметричной идентификации, которая получится при перестановке e, k: он требует, чтобы четвертый член был расположен в сумме после первого. Антецеденты выделены указателем "программа". Они выполняют только арифметические действия. Первый антецедент перечисляет все разложения коэффициента d в произведение двух целочисленных множителей f, h. Второй - независимо от этого перечисляет разложения a в произведение целочисленных g, j. Из соображений симметрии, можно ограничиться только случаем $g > 0$. Четвертый антецедент вычисляет величину $b - fj$ по уже известным значениям b, f, j и проверяет ее делимость на g, идентифицируя таким образом i. Наконец, пятый антецедент сравнивает значения $gh + fi$ и c. Данная последовательность действий автоматически определяется компилятором ГЕНОЛОГа. Соотношения для коэффициентов исходного и результирующих многочленов, представленные антецедентами, легко находятся при раскрывании скобок. Уровень срабатывания приема равен 3.

В этом же пункте имеется обобщение приема на случай, когда коэффициенты a, b, c, d - не целые числа, а произвольные алгебраические выражения. Эти коэффициенты обрабатываются нормализатором "видумножение", после чего реализуется процедура перечисления их "делителей", аналогичная процедуре перечисления целочисленных делителей заданного числа. Теорема приема имеет следующий вид:

$\forall_{abcdefghij}(fh = d \,\&\, gj = a \,\&\, gi = b - fj \,\&\, gh + fi - c = 0 \rightarrow ae^3 + be^2 + ce + d = (f + ge)(h + ie + je^2))$

Переменные a, b, c, e идентифицируются с произвольными выражениями; допускается обращение коэффициентов a, b, c в единицу. Указатели "группировка(x1)", "группировка(x2)", "группировка(x3)" обеспечивают идентификацию a, b, c путем группировок всех членов, соответственно, с e^3, e^2 и e. Первый и второй антецеденты обращаются к нормализатору "видумножение" для разложения на множители выражений a, d.

Чтобы пары f, h и g, j идентифицировались путем всевозможных разбиений на две части множителей выражений d, a, введены указатели "делители(x6 x8)", "делители(x7 x10)". Эти указатели и учитывающие их фрагменты компилятора понадобились только для данной серии приемов. Однако, здесь было предпринято небольшое обобщение - введен справочник "делители", указывающий по двуместной ассоциативно-коммутативной операции (в данном случае - операции "умножение") процедуру, выполняющую всевозможные разбиения "на две части" группы ее операндов. Это позволяет пользоваться уже имеющимися возможностями компилятора в новых ситуациях, дописывая при необходимости процедуры данного справочника. Правая часть третьего антецедента обрабатывается сначала нормализатором раскрывания скобок "стандплюс", а затем нормализатором "видумножение". Идентификация левой части и определение выражения i выполняются стандартным образом, не требуя дополнительных указателей. Наконец, для обработки левой части четвертого антецедента применяется нормализатор "стандплюс". Прием весьма трудоемок, хотя и необходим в отдельных задачах для получения разложения. Поэтому он имеет ограничитель трудоемкости, значительно ослабляемый при наличии комментария "максимум". Уровень срабатывания равен 5.

(b) Представление многочлена четвертой степени в виде произведения двух квадратных трехчленов.

$\forall_{abcdefghijklm}(gj = a \,\&\, il = f \,\&\, 0 < i \,\&\, hk = c-ij-gl \,\&\, hj+gk = b \,\&\, hl+ik = d \rightarrow am^4+bem^3+ce^2m^2+de^3m+fe^4 = (gm^2+hem+ie^2)(jm^2+kem+le^2))$

Коэффициенты a, b, c, d, f идентифицируются с целыми числами. Антецеденты выделены указателем "программа" - они выполняют арифметические действия. Первые два перечисляют разложения в произведение двух множителей коэффициентов a, f. Четвертый антецедент вычисляет величину $c - ij - gl$ и перечисляет всевозможные представления ее в виде произведения двух множителей h, k. Пятый и шестой - проверяют дополнительные условия на найденные параметры. Уровень срабатывания равен 3.

Как и в случае многочлена третьей степени, данный прием обобщен на случай нечисловых коэффициентов:

$\forall_{abcdfghijklm}(gj = a \,\&\, il = f \,\&\, hk = c-ij-gl \,\&\, hj+gk-b = 0 \,\&\, hl+ik-d = 0 \rightarrow am^4 + bm^3 + cm^2 + dm + f = (gm^2 + hm + i)(jm^2 + km + l))$

Идентификация переменных g, j, i, l, h, k определяется указателями "делители(x7 x10)", "делители(x9 x12)", "делители(x8 x11)". Прием срабатывает на уровне 5. Он весьма трудоемок, и для более высоких степеней его аналоги не созданы.

(c) Представление многочлена четвертой степени в виде произведения линейного множителя и многочлена третьей степени.

$\forall_{abcdefghijklm}(ad = h \,\&\, bf = l \,\&\, 0 < b \,\&\, bc = k-af \,\&\, be = j-ac \,\&\, bd+ae = i \rightarrow hg^4+img^3+jg^2m^2+kgm^3+lm^4 = (ag+bm)(dg^3+eg^2m+cgm^2+fm^3))$

Коэффициенты h, i, j, k, l идентифицируются с целыми числами. Первые два антецедента перечисляют разложения коэффициентов h, l; четвертый и пятый - однозначно доопределяют c, e. Шестой антецедент проверяет дополнительное условие на найденные параметры.

(d) Переход от многочлена с иррациональными коэффициентами к многочлену с целыми коэффициентами.

Несколько приемов выполняют замену переменной для исключения радикалов из коэффициентов. Таким образом подготавливается возможность применения приемов разложения многочленов с целыми коэффициентами.

$\forall_{abcdefghijklm}(j = ce \,\&\, k = fc \,\&\, l = gc^2 \,\&\, m = hc^2 \,\&\, a = b\sqrt{c} \,\&\, i = da^4 + ja^3 + ka^2 + la + m \rightarrow db^4 + e\sqrt{c}b^3 + fb^2 + g\sqrt{c}b + h = i/c^2)$

Переменные d, e, f, g, h идентифицируются с целыми числами; под радикалом расположена натуральная константа c. Переходя от переменной b к переменной $a = b\sqrt{c}$ и домножая на c^2, получаем многочлен i с целыми коэффициентами. К этому многочлену применяем нормализатор "видумножение", после чего делим результат на c^2. Указатель "свертка(x1 число(x1))" означает, что сначала многочлен i формируется относительно новой переменной X, а после применения к нему нормализаторов приема (в том числе нормализатора "видумножение") эта переменная заменяется на выражение $b\sqrt{c}$, заданное пятым антецедентом. Уровень срабатывания равен 3.

$\forall_{abcdefghij}(h = bd \;\&\; i = bf \;\&\; g = \sqrt{b}e \;\&\; j = ag^3 + cg^2 + hg + i \rightarrow a\sqrt{b}e^3 + ce^2 + d\sqrt{b}e + f = j/b)$

Аналогично предыдущему. В приведенных двух приемах радикалы имелись в коэффициентах нечетных степеней. Чтобы перейти к этому случаю от случая с радикалами при четных степенях, созданы еще два приема:

$\forall_{abcdefghi}(g = cd \;\&\; h = df \;\&\; i = a\sqrt{d}b^3 + gb^2 + e\sqrt{d}b + h \rightarrow ab^3 + c\sqrt{d}b^2 + eb + f\sqrt{d} = i/\sqrt{d})$

$\forall_{abcdefghijk}(h = ab \;\&\; i = be \;\&\; j = bg \;\&\; k = hc^4 + d\sqrt{b}c^3 + ic^2 + f\sqrt{b}c + j \rightarrow a\sqrt{b}c^4 + dc^3 + e\sqrt{b}c^2 + fc + g\sqrt{b} = k/\sqrt{b})$

Уровни срабатывания равны 3.

Начиная с многочленов пятой степени, применяется описываемый ниже прием, реализующий алгоритм Берлекэмпа. Это оказывается менее трудоемким, чем использование приемов, основанных на соображениях делимости коэффициентов. Однако, несколько таких приемов все же сохранены - они срабатывают для однородных многочленов от двух переменных.

(e) Представление многочлена пятой степени в виде произведения линейного множителя и многочлена четвертой степени.

$\forall_{abcdefghijklmnp}(jl = a \;\&\; ke = i \;\&\; 0 < k \;\&\; jm = b - kl \;\&\; jf = g - km \;\&\; jd = c - fk \;\&\; dk + je = h \rightarrow an^5 + bn^4p + gn^3p^2 + cn^2p^3 + hnp^4 + ip^5 = (jn + kp)(ln^4 + mn^3p + fn^2p^2 + dnp^3 + ep^4))$

Уровень срабатывания равен 3.

(f) Представление многочлена пятой степени в виде произведения многочленов второй и третьей степени.

$\forall_{abcdefpqrmnhsvwxy}(a = pm \;\&\; f = rs \;\&\; v = sp \;\&\; w = r^2m - pe \;\&\; vq^2 + wq = p^2sr + r^2b - drp \;\&\; np = b - qm \;\&\; hr = e - qs \;\&\; c = rm + qn + ph \;\&\; 0 < p \rightarrow ax^5 + bx^4y + cx^3y^2 + dx^2y^3 + exy^4 + fy^5 = (px^2 + qxy + ry^2)(mx^3 + nx^2y + hxy^2 + sy^3))$

Уровень срабатывания равен 4.

(g) Представление многочлена шестой степени в виде однородного многочлена третьей степени от квадратного двучлена и переменной.

$\forall_{abcdefghijklmnp}(hl^3 = g \;\&\; il^2 = f \;\&\; hm^3 = a \;\&\; im^2 = b \;\&\; k = d - 2lmi \;\&\; jl = e - 3hl^2m \;\&\; 3lm^2h + mj = c \;\&\; n = h(l + mp^2)^3 + ip(l + mp^2)^2 + jp^2(l + mp^2) + kp^3 \rightarrow ap^6 + bp^5 + cp^4 + dp^3 + ep^2 + fp + g = n)$

Этот прием несколько отличается от предыдущих. Он предпринимает попытку понизить размерность задачи - усмотреть в многочлене шестой степени однородный многочлен третьей степени от квадратного двучлена $mp^2 + l$ и переменной p. После этого реализуется обращение к нормализатору "видумножение" для разложения на множители многочлена третьей степени (это делает последний антецедент). Коэффициенты a, b, c, d, e, f, g идентифицируются с целочисленными константами. Первые семь антецедентов, выделенные указателями "программа", выполняют арифметические вычисления, необходимые для определения коэффициентов h, i, j, k многочлена третьей степени. Первый антецедент перечисляет разложения параметра g в произведение двух множителей, причем отбираются лишь те из них, где второй множитель - куб некоторого целого числа l. Второй, третий и пятый антецеденты последовательно доопределяют коэффициенты

i, m, k. Четвертый и шестой антецеденты проверяют дополнительные соотношения, необходимые для возможности перехода к многочлену третьей степени. Уровень срабатывания приема равен 4.

(h) Представление многочлена четвертой степени в виде однородного многочлена второй степени от переменной и квадратного двучлена.

$\forall_{abcdefghijku}(e = fj^2 \;\&\; a = fk^2 \;\&\; d = gj \;\&\; b = gk \;\&\; h = c - 2jkf \;\&\; u = \lambda_x(fx^2 + gx + h, x - \text{число}) \to ai^4 + bi^3 + ci^2 + di + e = i^2u(j + ki^2/i))$

Первые пять антецедентов выделены указателями "программа"; они выполняют арифметические вычисления, необходимые для определения коэффициентов f, g, h однородного многочлена второй степени от двучлена $j + ki^2$ и переменной i. Последний антецедент обращается к нормализатору "видумножение" для разложения квадратного трехчлена $fx^2 + gx + h$ на множители. Чтобы можно было подставить в результат разложения отношение $(j + ki^2)/i$ вместо переменной x, он оформляется как вспомогательная функция $u(x)$, а переменная u выделяется указателем "отображение". Прием разлагает многочлен в произведение квадратных трехчленов, имеющих, вообще говоря, нецелочисленные коэффициенты - могут возникать новые числовые радикалы. Поэтому его применение сильно ограничено: фактически он был нужен лишь при формальном интегрировании, что и нашло свое отражение в фильтрах. Уровень срабатывания равен 4.

(i) Представление многочлена шестой степени в виде произведения линейного множителя и многочлена пятой степени.

$\forall_{abcdefghkijnplmrs}(ih = a \;\&\; js = g \;\&\; 0 < j \;\&\; ik = b - jh \;\&\; il = c - kj \;\&\; im = d - jl \;\&\; ir = e - jm \;\&\; is + jr = f \to an^6 + bn^5p + cn^4p^2 + dn^3p^3 + en^2p^4 + fnp^5 + gp^6 = (in + jp)(hn^5 + kn^4p + ln^3p^2 + mn^2p^3 + rnp^4 + sp^5))$

Уровень срабатывания равен 4.

(j) Представление многочлена шестой степени в виде произведения квадратного трехчлена и многочлена четвертой степени.

$\forall_{abcdefgmnpxyABCDEk}(mA = a \;\&\; pE = g \;\&\; nk = m(pmd - m^2f - p^2b) \;\&\; b - nA = mB \;\&\; f - nE = pD \;\&\; c - pA - nB = mC \;\&\; mD + nC + pB = d \;\&\; mE + Dn + Cp = e \to ax^6 + bx^5y + cx^4y^2 + dx^3y^3 + ex^2y^4 + fxy^5 + gy^6 = (mx^2 + nxy + py^2)(Ax^4 + Bx^3y + Cx^2y^2 + Dxy^3 + Ey^4))$

Прием срабатывает на уровне 4. Вычисления здесь уже становятся достаточно трудоемкими, и оставшиеся несколько приемов разложения на множители из соображений делимости коэффициентов, реализованные на ГЕНОЛОГе, не откомпилированы. Сюда относятся: представление многочленов шестой степени в виде произведения двух многочленов третьей степени и разложение на множители многочленов седьмой степени. В этих случаях существенно быстрее работает реализованный на ГЕНОЛОГе алгоритм Берлекэмпа.

10. Сложение числовых констант. При развитии нормализатора обнаруживались ситуации, в которых промежуточное выражение требовало дополнительной простейшей стандартизации. Для них создано несколько приемов, например, прием, выполняющий сложение числовых констант:

$\forall_{abcd}(c = a + b \to a + b + d = c + d)$

Переменные a, b идентифицируются с десятичными константами. Прием срабатывает на уровне 1.

11. Вспомогательное раскрывание скобок. Во многих случаях успешное разложение на множители требует предварительного раскрывания скобок. Поэтому на уровне 5 срабатывает прием, предпринимающий попытку раскрыть скобки и после этого разложить на множители:

    $\forall_{ab}(b = a \to a = b)$

    Антецедент обращается сначала к нормализатору "стандплюс", затем - к нормализатору "видумножение". Чтобы заблокировать попытку повторного раскрывания скобок в последнем обращении, используется комментарий "станд". Проверяется, что полученное выражение b не совпадает с исходным выражением a. Указатель "выход" обеспечивает немедленную выдачу b в качестве окончательного результата (так как внутреннее обращение к нормализатору "видумножение" уже обеспечило применение всех доступных средств, продолжение реализации внешнего обращения избыточно). Перед применением приема проверяется, что a не имеет мультипликативного заголовка ("умножение", "дробь", "степень", "минус") и что существует некорневое вхождение суммы в a. Для особого случая, когда любые два слагаемых сумм внутри a не имеют общего параметра, и это же верно для любых двух сомножителей произведений в a, создан отдельный прием. Здесь успешное разложение на множители настолько маловероятно, что применяется только раскрывание скобок, и сразу выдается результат.

12. Выделение множителей из дробной степени.

    Если числитель дробного показателя степени больше знаменателя, а в основании степени находится произведение, то выделяется целая часть показателя:

    $$\forall_{abcdefg}\left(0 < b - c \;\&\; b = cd + e \;\&\; 0 \le ah \to (ah)^{\frac{b}{c}} f + g = a^d h^d (ah)^{\frac{e}{c}} f + g\right)$$

    Переменные b, c идентифицируются с натуральными константами, причем c четное. Первые два антецедента выделены указателями "программа" (второй выполняет деление с остатком). Неполное частное d не более 3. Сомножители a, h не являются суммами.

13. Вынесение за скобку общего множителя экспоненциального вида.

    Если каждое слагаемое имеет степенной множитель с заданным основанием f, причем хотя бы один показатель степени - неконстантный, то выбирается степенной множитель f^e с минимальным (в уточняем ниже эвристическом смысле)e, и он выносится за скобку:

    $\forall_{abcdef}(af^e + bf^d + c = f^e(a + bf^{d-e} + cf^{-e}))$

    Выражение d неконстантное. Каждое слагаемое остаточной суммы c имеет сомножитель вида f^n, у которого показатель n имеет общий параметр с показателем e. Под "минимальностью" показателя e понимается, во-первых, приоритетный выбор константного показателя (это возможно лишь при нулевом c), и, во-вторых, отсутствие такого другого показателя m, что выражение $m - e$

после упрощения приобретает внешний знак "минус". Уровень срабатывания равен 2.

Дополнительно рассмотрен случай, когда основание степени - сумма a, образующая подсумму разлагаемого на множители выражения:

$\forall_{abcd}(0 < a \rightarrow a + a^b c + d = a(1 + a^{b-1}c + da^{-1}))$

Каждое слагаемое выражения d имеет сомножитель - степень с основанием a. Показатель степени b должен содержать неизвестные; если d отлично от 0, то и основание a должно быть не известным. Уровень срабатывания равен 2.

14. Действия с логарифмами. Несколько несложных приемов преобразуют логарифмические выражения: сводят их к более простым либо к уже имевшимся в преобразуемом терме. Таким образом повышается вероятность успешного разложения на множители.

    (a) Попытка разложения на множители суммы под логарифмом.

    $\forall_{abcdefg}(d = b + c \ \& \ 0 < b + c \ \& \ e - \text{rational} \ \& \ \neg(\text{знаменатель}(e) - \text{even}) \rightarrow f \cdot (\log_a(b + c))^e + g = f \cdot (\log_a d)^e + g)$

    Первый антецедент обращается для разложения на множители суммы $b+c$ к нормализатору "видумножение". Проверяется, что результат d отличается от этой суммы. Уровень обращения равен 1.

    (b) Логарифм произведения.

    $\forall_{abcdef}(0 \leq a \ \& \ 0 \leq b \rightarrow d(\log_c(ab))^g + e = d(\log_c a + \log_c b)^g + e)$

    Переменная g идентифицируется с простой дробью, имеющей нечетный знаменатель. Уровень срабатывания равен 1.

    (c) Логарифм дроби.

    $\forall_{abcdefg}(0 \leq a \ \& \ 0 \leq b \rightarrow d(\log_c(a/b))^g + e = d(\log_c a - \log_c b)^g + e)$

    Аналогично предыдущему.

    (d) Переход к другому основанию.

    $\forall_{abcdef}(0 < a \ \& \ \neg(a - 1 = 0) \ \& \ 0 < b \ \& \ \neg(b - 1 = 0) \ \& \ e - \text{rational} \ \& \ \neg(\text{знаменатель}(e) - \text{even}) \rightarrow d(\log_b c)^e + f = d(\log_a c)^e/(\log_a b)^e + f)$

    Выражения $\log_a b$, $\log_a c$ до преобразования уже имеются в обрабатываемом нормализатором терме. Уровень срабатывания равен 2.

15. Возведение константы в натуральную степень. Если преобразуемое выражение имеет вид натуральной степени десятичной константы, то находится его численное значение:

    $\forall_{abc}(a^b = c \rightarrow a^b = c)$

    Переменная a идентифицируется с десятичной константой, переменная b - с натуральным числом, не превосходящим 5. Для выполнения арифметических действий антецедент обращается к нормализатору "нормстепень". Комментарий "натуральное", используемый при разложении на множители натуральных констант, блокирует применение приема. Уровень срабатывания равен 1.

16. Разложение на два множителя натурального основания степени.

    $\forall_{abcdef}(b = ef \rightarrow ab^c + d = ae^c f^c + d)$

Переменная b идентифицируется с натуральной константой. Для идентификации e усматривается слагаемое преобразуемого выражения, имеющее натуральную константу e своим основанием степени (такое добавление к идентификации вынесено в указатель "контекст(...)"). Антецедент выделен указателем "программа" - он выполняет деление b на e. Показатель степени должен быть неконстантным. Если число слагаемых меньше 4, то прием блокируется. Уровень срабатывания равен 2.

17. Раскрывание скобок в показателе степени.

$$\forall_{abcdefghp}\left(p = (c+d)^e f + g \to ab^{(c+d)^e f+g} + h = ab^p + h\right)$$

Раскрывание скобок в показателе степени относится к числу преобразований общей стандартизации, применяемой решателем. Прием инициируется, если сомножитель слагаемого показателя степени имеет вид суммы либо натуральной степени суммы. Переменная e идентифицируется с натуральной константой, не большей 3. Либо e, либо f должно быть отлично от единицы. Раскрывание скобок выполняет антецедент, обращающийся к нормализатору "стандплюс". Уровень срабатывания равен 2.

18. Разложение на множители тригонометрических выражений. Это очень большой раздел нормализатора "видумножение", в которой вошла значительная часть тригонометрических тождеств. Он разбивается на следующие подразделы:

    (a) Непосредственное разложение на множители. Перечислим приемы, позволяющие за один шаг перейти к мультипликативному тригонометрическому выражению:

        i. Сумма тригонометрической операции и константного терма.
        Приемы, в которых после деления на $2b$ возникает сумма либо разность с синусом $\pi/6$:
        $\forall_{ab}(b - 2b\cos a = 4b\sin(a/2+\pi/6)\sin(a/2-\pi/6))$
        $\forall_{ab}(2b\sin a + b = 4b\sin(a/2+\pi/12)\cos(a/2-\pi/12))$
        $\forall_{ab}(b + 2b\cos a = 4b\cos(a/2+\pi/6)\cos(a/2-\pi/6))$
        $\forall_{ab}(2b\sin a - b = 4b\cos(a/2+\pi/12)\sin(a/2-\pi/12))$
        Уровень срабатывания равен 1.
        Приемы, в которых после деления на $\sqrt{2}b$ возникает сумма либо разность с синусом $\pi/4$:
        $\forall_{ab}(b + \sqrt{2}b\cos a = 2\sqrt{2}b\cos(a/2+\pi/8)\cos(a/2-\pi/8))$
        $\forall_{ab}(b - \sqrt{2}b\cos a = 2\sqrt{2}b\sin(a/2+\pi/8)\sin(a/2-\pi/8))$
        $\forall_{ab}(\sqrt{2}b\sin a + b = 2\sqrt{2}b\sin(a/2+\pi/8)\cos(a/2-\pi/8))$
        $\forall_{ab}(\sqrt{2}b\sin a - b = 2\sqrt{2}b\cos(a/2+\pi/8)\sin(a/2-\pi/8))$
        Уровень срабатывания равен 2.
        Приемы, в которых после деления на $\sqrt{3}b$ возникает сумма либо разность с синусом $\pi/3$:
        $\forall_{ab}(\sqrt{3}b - 2b\sin a = 4b\sin(\pi/6-a/2)\cos(\pi/6+a/2))$
        $\forall_{ab}(\sqrt{3}b + 2b\sin a = 4b\sin(a/2+\pi/6)\cos(\pi/6-a/2))$
        Уровень срабатывания равен 1.

Во всех перечисленных приемах требовалось отсутствие комментария "числитель", вводимого при разложении на множители числителя либо знаменателя, содержащего тригонометрические операции. Следующие два приема лишены этого ограничения; они используют переход к половинному аргументу, если во внешнем контексте уже имелись множители, сокращаемые с новыми множителями либо выносимые вместе с ними за скобку:
$\forall_{abcd}(b = a/2 \ \& \ c = 2\cos b - 1 \ \& \ d = 2\cos b + 1 \rightarrow 1 + 2\cos a = cd)$
$\forall_{abd}(d = \sin(a/2) \rightarrow b - b\cos a = 2bd^2)$
Уровень срабатывания равен 4.

ii. Сумма квадрата тангенса либо котангенса и константы.

$$\forall_a \left((\operatorname{tg} a)^2 - 3 = -\frac{4\cos(a + \pi/6)\cos(a - \pi/6)}{(\cos a)^2} \right)$$

$$\forall_a \left(3(\operatorname{tg} a)^2 - 1 = \frac{4\sin(a - \pi/6)\sin(a + \pi/6)}{(\cos a)^2} \right)$$

$$\forall_a \left(3(\operatorname{ctg} a)^2 - 1 = \frac{4\cos(a - \pi/6)\cos(a + \pi/6)}{(\sin a)^2} \right)$$

Уровень срабатывания равен 1.

iii. Разность синусов.

$$\forall_{abcde} \left(d = \cos\frac{a+b}{2} \ \& \ e = \sin\frac{a-b}{2} \rightarrow c\sin a - c\sin b = 2cde \right)$$

Антецеденты обрабатывают множители d, e нормализаторами общей стандартизации. Если преобразуемое выражение константное, причем a либо b содержит обратную тригонометрическую функцию, то прием блокируется. Уровень срабатывания равен 1.

iv. Сумма синусов.

$$\forall_{abcde} \left(d = \cos\frac{a-b}{2} \ \& \ e = \sin\frac{a+b}{2} \rightarrow c\sin a + c\sin b = 2cde \right)$$

v. Сумма косинусов.

$$\forall_{abcde} \left(d = \cos\frac{a+b}{2} \ \& \ e = \cos\frac{a-b}{2} \rightarrow c\cos a + c\cos b = 2cde \right)$$

vi. Разность косинусов.

$$\forall_{abcde} \left(d = \sin\frac{a+b}{2} \ \& \ e = \sin\frac{a-b}{2} \rightarrow c\cos a - c\cos b = -2cde \right)$$

vii. Сумма синуса и косинуса.
$\forall_{abcde}(d = \sin(\pi/4 + a/2 - b/2) \ \& \ e = \cos(a/2 + b/2 - \pi/4) \rightarrow c\sin a + c\cos b = 2cde)$

В этом случае преобразование дает более сложные множители, чем предыдущие преобразования, и появляются дополнительные фильтры. Если выражение содержит неизвестные, то оба аргумента a, b должны быть неизвестными. При интегрировании прием блокируется. При наличии комментария "уравнменьше" (разложение на множители разности частей неравенства) проверяется, что один из аргументов не равен половине другого.

Дополнительно созданы два приема для разложения на множители трехчленов, в которых преобразование суммы синуса и косинуса приводит к двучлену для суммы тригонометрических функций:

$\forall_{abc}(c\sqrt{3}\cos a + c\sin a + b = 2c\cos(a - \pi/6) + b)$

$\forall_{abc}(c\sqrt{3}\sin a + c\cos a + b = 2c\sin(a + \pi/6) + b)$

В обоих случаях проверяется, что b либо равно 0, либо имеет вид $2cd$, где заголовком d служит синус или косинус.

viii. Разность синуса и косинуса.

Приемы аналогичны случаю суммы:

$\forall_{abcde}(d = \cos(a/2 - b/2 + \pi/4)\ \&\ e = \sin(a/2 + b/2 - \pi/4) \rightarrow c\sin a - c\cos b = 2cde)$

$\forall_{abc}(\sqrt{3}c\sin a - c\cos a + b = 2c\sin(a - \pi/6) + b)$

$\forall_{abc}(\sqrt{3}c\cos a - c\sin a + b = 2c\cos(a + \pi/6) + b)$

ix. Синус двойного аргумента.

$\forall_{abc}(a = \sin(b/2)\ \&\ c = \cos(b/2) \rightarrow \sin b = 2ac)$

Прием применяется только при наличии комментария "числитель". Анализируется внешний контекст, и проверяется, что при замене либо появляется возможность сократить внешнюю дробь на один из множителей a,c, либо эта дробь является слагаемым в сумме нескольких дробей, уже содержащих a,c среди множителей своих знаменателей или числителей (соответственно тому, относилось ли текущее вхождение к знаменателю или числителю). Уровень срабатывания равен 3.

x. Косинус двойного аргумента. В зависимости от внешнего контекста, выбирается одна из двух возможностей - переход к произведению суммы синуса и косинуса на их разность, либо переход к удвоенному произведению синуса на косинус:

$\forall_{abc}(b = \cos(a/2) - \sin(a/2)\ \&\ c = \cos(a/2) + sin(a/2) \rightarrow \cos a = bc)$

$\forall_{abc}(b = \sin(a/2 + \pi/4)\ \&\ c = \cos(a/2 + \pi/4) \rightarrow \cos a = 2bc)$

В каждом из случаев проверяется, что выполнение преобразования либо дает возможность сократить внешнюю дробь, либо выявляет общий множитель знаменателей двух дробных слагаемых внешней суммы. Уровень срабатывания равен 3.

xi. Переход к синусу либо косинусу суммы либо разности. Если это возможно, преобразуемая сумма сворачивается по формулам синуса либо косинуса суммы (разности):

$\forall_{ab}(\sin a\cos b + \sin b\cos a = \sin(a + b))$

$\forall_{ab}(\sin a\cos b - \cos a\sin b = \sin(a - b))$

$\forall_{ab}(\sin a\sin b + \cos a\cos b = \cos(a - b))$

$\forall_{ab}(\cos a\cos b - \sin a\sin b = \cos(a + b))$

Дополнительные слагаемые отсутствуют. Приемы срабатывают на уровне 1.

xii. Усмотрение множителя - суммы либо разности синуса и косинуса. Если косинус двойного аргумента добавляется к сумме либо разности синуса и косинуса, то последнюю можно вынести за скобки:

$\forall_{abcd}(d = 2c \to a\cos d + b\sin c + b\cos c = (\cos c + \sin c)(b + a\cos c - a\sin c))$
$\forall_{abcd}(d = 2c \to a\cos d + b(\sin c + \cos c) = (\cos c + \sin c)(b + a\cos c - a\sin c))$
$\forall_{abcd}(d = 2c \to a\cos d + b\cos c - b\sin c = (\cos c - \sin c)(b + a\cos c + a\sin c))$
В первом и третьем приемах блокируется ситуация $a = \pm\sqrt{2}b$, так как здесь возможно альтернативное разложение, использующее вспомогательный угол $\pi/4$. Уровень срсбатывания приемов равен 3.

xiii. Усмотрение множителя - синуса либо косинуса от суммы с $\pi/4$.
$\forall_{abc}(a\cos c - a\sin c + b\cos(c + \pi/4) = \cos(c + \pi/4)(b + \sqrt{2}a))$
$\forall_{abc}(a\sin c + a\cos c + b\sin(c + \pi/4) = \sin(c + \pi/4)(b + \sqrt{2}a))$
$\forall_{abc}(a(\cos c - \sin c) + b\cos(c + \pi/4) = \cos(c + \pi/4)(b + \sqrt{2}a))$
$\forall_{abc}(a(\sin c + \cos c) + b\sin(c + \pi/4) = \sin(c + \pi/4)(b + \sqrt{2}a))$
$\forall_{abcd}(d = 2c \to a\cos d + b\sin(c + \pi/4) = \sin(c + \pi/4)(b + a\sqrt{2}\cos c - a\sqrt{2}\sin c))$
Уровень срабатывания равен 3.

xiv. Усмотрение множителя - синуса либо косинуса от суммы с $\pi/6$.
$\forall_{abcde}(c - \sqrt{3}\sin b = 0 \;\&\; e - \cos b = 0 \to a\sin(-b + \pi/6) - cd + ed = \sin(-b + \pi/6)(a + 2d))$
$\forall_{abcde}(c - \sqrt{3}\sin b = 0 \;\&\; e - \cos b = 0 \to a\sin(b + \pi/6) + cd + ed = \sin(b + \pi/6)(a + 2d))$
Антецеденты выделены указателями "идентификатор". Коэффициент a, выделенный указателем "группировка", может складываться из нескольких слагаемых с общим множителем $\sin(\pm b + \pi/6)$. Уровень срабатывания равен 1.

xv. Произведение тангенсов либо котангенсов плюс-минус единица.

$$\forall_{abc}\left(c\,\mathrm{tg}\,a\,\mathrm{tg}\,b - c = -\frac{c\cos(a+b)}{\cos a\cos b}\right)$$

$$\forall_{abc}\left(c\,\mathrm{tg}\,a\,\mathrm{tg}\,b + c = \frac{c\cos(a-b)}{\cos a\cos b}\right)$$

$$\forall_{abc}\left(c\,\mathrm{tg}\,a\,\mathrm{ctg}\,b + c = \frac{c\sin(a+b)}{\cos a\sin b}\right)$$

$$\forall_{abc}\left(c\,\mathrm{tg}\,a\,\mathrm{ctg}\,b - c = \frac{c\sin(a-b)}{\cos a\sin b}\right)$$

$$\forall_{abc}\left(c\,\mathrm{ctg}\,a\,\mathrm{ctg}\,b - c = \frac{c\cos(a+b)}{\sin a\sin b}\right)$$

$$\forall_{abc}\left(c\,\mathrm{ctg}\,a\,\mathrm{ctg}\,b + c = \frac{c\cos(a-b)}{\sin a\sin b}\right)$$

Уровень срабатывания равен 1.

xvi. Шестая степень синуса либо косинуса минус единица.
$\forall_{ab}(a - a(\sin b)^6 = a(\cos b)^2(1 + (\sin b)^2 + (\sin b)^4))$
$\forall_{ab}(a - a(\cos b)^6 = a(\sin b)^2(1 + (\cos b)^2 + (\cos b)^4))$
Уровень срабатывания равен 2.

xvii. Линейная комбинация синуса (косинуса) и синуса двойного аргумента.
$\forall_{abcd}(d = 2c \to a\sin d + b\cos c = \cos c(b + 2a\sin c))$
$\forall_{abcd}(d = 2c \to a\sin d + b\sin c = \sin c(b + 2a\cos c))$

Проверяется, что выражение не может быть свернуто по формуле синуса (косинуса) суммы либо разложено на множители по формуле суммы тригонометрических операций. Уровень срабатывания равен 2.

xviii. Линейная комбинация синуса и синуса учетверенного аргумента.
$\forall_{abcd}(d = 4c \to a\sin d + b\sin c = \sin c(b + 4a\cos c\cos(2c)))$
Ограничения на применение приема - те же, что в предыдущем пункте. Уровень срабатывания равен 3.

xix. Преобразование разности квадратов косинуса и синуса к косинусу двойного аргумента.
$\forall_a((\cos a)^2 - (\sin a)^2 = \cos(2a))$
Проверяется, что внешний контекст не содержит указаний на целесообразность альтернативного разложения - в произведение суммы и разности синуса и косинуса. Уровень срабатывания равен 1.

xx. Сумма трех косинусов.
$\forall_{abcde}(e = \sin\frac{a-c}{2}\sin\frac{a+c}{2}+\sin\frac{b-c}{2}\sin\frac{b+c}{2} \to d\cos a+d\cos b-2d\cos c = 2de)$
Антецедент предпринимает попытку разложить сумму e на множители, используя обращение к нормализатору "видумножение". При этом используется комментарий "тригонометрия", блокирующий преобразования произведений тригонометрических операций в суммы. Прием применяется только при успехе данной попытки. Уровень срабатывания равен 2.

xxi. Единица минус квадрат тангенса.

$$\forall_a\left(1 - (\operatorname{tg} a)^2 = \frac{\cos(2a)}{(\cos a)^2}\right)$$

Выражение a должно содержать неизвестные. Уровень срабатывания равен 1.

xxii. Квадрат секанса минус единица.
$\forall_a((\sec a)^2 - 1 = (\operatorname{tg} a)^2)$
Уровень срабатывания равен 1.

xxiii. Усмотрение множителя вида $1 \pm \sin x$ либо $1 \pm \cos x$.
$\forall_{abxkpq}(0 \leq k-3 \ \&\ q-p = 1 \to a(\cos x)^k + b(\sin x)^p + b(\sin x)^q = (1+\sin x)(b(\sin x)^p + a(\cos x)^{k-2}(1-\sin x)))$
$\forall_{abxkpq}(0 \leq k-3 \ \&\ q-p = 1 \to a(\cos x)^k + b(\sin x)^p - b(\sin x)^q = (1-\sin x)(b(\sin x)^p + a(\cos x)^{k-2}(1+\sin x)))$
Переменные k, p, q идентифицируются с натуральными константами. Указатель "альтернатива(...)" предусматривает возможность одновременной замены синусов на косинусы и наоборот. Уровень срабатывания равен 2.

(b) Группировки для тригонометрических операций.

В случае тригонометрических операций добавляются следующие приемы разложения на множители путем частичной группировки слагаемых:

i. Разность синусов.
$\forall_{abcdefg}(d = \cos\frac{a+b}{2} \ \&\ e = \sin\frac{a-b}{2} \ \&\ f = 2cde+g \to c\sin a-c\sin b+g = f)$
Первый и второй антецеденты выполняют общую стандартизацию множителей d, e, третий - обращается к нормализатору "видумножение"

для разложения на множители выражения, полученного при группировке. Обращение сопровождается комментарием "группировка", блокирующим преобразования перехода от произведений тригонометрических операций к суммам и раскрывание скобок. Созданы две версии приема; первая срабатывает на уровне 2, вторая - на уровне 3. Обе они проверяют выполнение следующих условий:

A. После попытки разложения на множители выражение f либо не имеет вида суммы (быть может, с вынесенным наружу знаком "минус"), либо имеет единственный содержащий неизвестные тригонометрический аргумент.

B. Если есть комментарий "группировка", то должно существовать слагаемое остаточной суммы g, делящееся на один из новых множителей d, e.

C. Общее число слагаемых преобразуемого выражения не более 8.

D. Преобразуемое выражение неконстантное.

E. Остаточная сумма g не имеет дробных слагаемых.

F. Число содержащих неизвестные тригонометрических аргументов преобразуемого выражения не равно 1.

G. Либо остаточная сумма g имеет хотя бы один общий параметр с выражениями a, b либо она не содержит неизвестных, и один из новых множителей d, e тоже не содержит неизвестных.

Первый прием требует дополнительного выполнения следующих условий:

A. Преобразуемое выражение содержит неизвестные.

B. Хотя бы один из новых множителей либо не содержит неизвестных, либо делит слагаемое остаточной суммы g. В последнем случае число слагаемых данной суммы не менее 5.

C. Либо число слагаемых преобразуемого выражения не менее 5, либо хотя бы один из аргументов a, b меет заголовок "плюс".

Указатель "выход" в случае успешного применения одного из данных приемов обрывает дальнейшее применение нормализатора и инициирует немедленную выдачу ответа.

ii. Сумма синусов.
$\forall_{abcdefg}(d=\cos\frac{a-b}{2}\ \&\ e=\sin\frac{a+b}{2}\ \&\ f=2cde+g\rightarrow c\sin a+c\sin b+g=f)$
Этот и следующие два пункта аналогичны предыдущему.

iii. Сумма косинусов.
$\forall_{abcdefg}(d=\cos\frac{a+b}{2}\ \&\ e=\cos\frac{a-b}{2}\ \&\ f=2cde+g\rightarrow c\cos a+c\cos b+g=f)$

iv. Разность косинусов.
$\forall_{abcdefg}(d=\sin\frac{a+b}{2}\ \&\ e=\sin\frac{a-b}{2}\ \&\ f=-2cde+g\rightarrow c\cos a-c\cos b+g=f)$

v. Сумма синуса и косинуса.
$\forall_{abcdefg}(d=\sin(a/2-b/2+\pi/4)\ \&\ e=\cos(a/2+b/2-\pi/4)\ \&\ f=2cde+g\rightarrow c\sin a+c\cos b+g=f)$

В этом случае создана единственная версия приема, срабатывающая на уровне 4. Фильтры аналогичны предыдущим приемам, с небольшим усилением: в первом фильтре вместо требования, чтобы результат не имел вида суммы, берется требование, чтобы он не имел вида суммы, домноженной на константный множитель. Кроме того, требуется отсутствие комментария "числитель".

vi. Разность синуса и косинуса.
$\forall_{abcdefg}(d = \cos(a/2 - b/2 + \pi/4)\ \&\ e = \sin(a/2 + b/2 - \pi/4)\ \&\ f = 2cde + g \to c\sin a - c\cos b + g = f)$
Аналогично предыдущему.

vii. Синус двойного аргумента.
$\forall_{abcdef}(d = \sin(b/2)\ \&\ e = \cos(b/2)\ \&\ f = 2ade + c \to a\sin b + c = f)$
Проверяется, что остаточная сумма c имеет слсгаемое, делящееся на d либо e. Прочие фильтры аналогичны предыдущим случаям. Уровень срабатывания равен 4.

viii. Сумма тригонометрической операции и константы.
$\forall_{abcdef}(d = \cos(a/2 + \pi/6)\ \&\ e = \cos(a/2 - \pi/6)\ \&\ f = 4bde + c \to b + 2b\cos a + c = f)$
$\forall_{abcdef}(d = \sin(a/2 + \pi/6)\ \&\ e = \sin(a/2 - \pi/6)\ \&\ f = 4bde + c \to b - 2b\cos a + c = f)$
$\forall_{abcdef}(d = \cos(a/2 + \pi/12)\ \&\ e = \sin(a/2 - \pi/12)\ \&\ f = c - 4bde \to b - 2b\sin a + c = f)$
$\forall_{abcdef}(d = \sin(a/2 + \pi/12)\ \&\ e = \cos(a/2 - \pi/12)\ \&\ f = c + 4bde \to b + 2b\sin a + c = f)$
$\forall_{abcde}(d = \cos(a/2)\ \&\ e = 2bd^2 + c \to b + b\cos a + c = e)$
$\forall_{abcde}(d = \sin(a/2)\ \&\ e = 2bd^2 + c \to b - b\cos a + c = e)$
$\forall_{abcde}(d = \sin(a/2 + \pi/4)\ \&\ e = 2bd^2 + c \to b + b\sin a + c = e)$
$\forall_{abcde}(d = \sin(a/2 - \pi/4)\ \&\ e = 2bd^2 + c \to b - b\sin a + c = e)$
$\forall_{abcde}(d = \cos(a/2)\ \&\ e = 2bd^2 + c \to b + b\cos a + c = e)$
$\forall_{abcde}(d = \sin(a/2)\ \&\ e = 2bd^2 + c \to b - b\cos a + c = e)$
Последние два приема срабатывают на уровне 2, остальные приемы - на уровне 3. Фильтры аналогичны предыдущим случаям.

ix. Частичная свертка двух произведений в синус разности.
$\forall_{abcdefg}(f = c\sin((a-b)e) + (c+d)\cos(ae)\sin(be) + g \to c\sin(ae)\cos(be) + d\cos(ae)\sin(be) + g = f)$
Переменные c, d идентифицируются с десятичными константами; a - с натуральной константой; b - с четной константой. Выражение e содержит неизвестные. Для отсечения симметричных случаев проверяется, что величина a либо нечетная, либо меньше b. Антецедент обращается к нормализатору "видумножение". При наличии комментария "группировка", указывающего, что часть слагаемых уже была сгруппирована, прием блокируется. Уровень срабатывания равен 5. Так как никаких условий на результат f не накладывается, прием не имеет указателя "выход", обрывающего работу нормализатора.

x. Группировка по формуле синуса либо косинуса суммы.
$\forall_{abcde}(e = \sin(a + \pi/6)\ \&\ d = 2be + c \to \sqrt{3}b\sin a + b\cos a + c = d)$
$\forall_{abcde}(e = \sin(a - \pi/6)\ \&\ d = 2be + c \to \sqrt{3}b\sin a - b\cos a + c = d)$
$\forall_{abcde}(e = \sin(a + \pi/3)\ \&\ d = 2be + c \to \sqrt{3}b\cos a + b\sin a + c = d)$
$\forall_{abcde}(e = \sin(a - \pi/3)\ \&\ d = 2be + c \to -\sqrt{3}b\cos a + b\sin a + c = d)$
Уровень срабатывания приемов равен 3.

(c) Предварительная стандартизация тригонометрических множителей. Нормализатор использует большое число стандартизирующих тригонометрических преобразований, ориентированных на повышение вероятности получения разложения:

i. Изменение знака разности под синусом.
$\forall_{abcpq}(c = a - b \to p\sin(b-a) + q = q - p\sin c)$
Прием изменяет знак разности, если это дает выражение c, лексикографически предшествующее выражению $b - a$. Исключение составляет случай, когда исходная разность имеет своим слагаемым выражение вида $n\pi/m$ для натуральных m, n. Антецедент преобразует разность $a-b$, используя нормализаторы общей стандартизации и нормализатор "стандупорядочение", лексикографически упорядочивающий операнды коммутативных операций. Уровень срабатывания равен 1.

ii. Изменение знака разности под косинусом.
$\forall_{abcpq}(c = a - b \to p\cos(b-a) + q = q + p\cos c)$
Аналогично предыдущему.

iii. Переход к косинусу двойного аргумента.
$\forall_{abc}(2a(\cos b)^2 - a + c = a\cos(2b) + c)$
$\forall_{abc}(a - 2a(\sin b)^2 + c = a\cos(2b) + c)$
Либо b не содержит неизвестных, либо преобразуемое выражение имеет тригонометрическую операцию, отличную от $\cos b$ в первом приеме и от $\sin b$ во втором, либо a, c не содержат тригонометрических операций с неизвестными. Уровень срабатывания равен 2.
$\forall_{abc}(a(\cos b)^2 - a(\sin b)^2 + c = a\cos(2b) + c)$
Этот прием срабатываетя на уровне 1. Требуется существование в преобразуемом выражении тригонометрического аргумента, отличного от b.

Для блокировки перечисленных приемов используются комментарии "тригаргумент" и "фильтрудвоения".

iv. Выражение тангенса через синус и косинус.

Если выражение не составлено из тангенсов и котангенсов одного и того же тригонометрического аргумента, то тангенсы и котангенсы при разложении на множители выражаются через синусы и косинусы:
$\forall_{abcd}(b - \text{rational}\ \&\ \neg(\text{знаменатель}(b) - \text{even})\ \&\ \neg(\cos d = 0) \to a(\operatorname{tg} d)^b + c = a(\sin d)^b/(\cos d)^b + c)$
Для применения приема необходимо, чтобы либо преобразуемое выражение, либо выражение из внешнего контекста имело вхождение тригонометрической операции, аргумент которой отличался бы от d, либо заголовком служил синус или косинус. Уровень срабатывания равен 2. В особых случаях вместо данного приема используются приемы, выражающие тангенс через синус и косинус двойного аргумента:
$\forall_{abcdef}(c = 1 + \cos(2a)\ \&\ d = \sin(2a)\ \&\ \neg(\cos a = 0) \to e(\operatorname{tg} a)^b + f = ed^b/c^b + f)$
$\forall_{abcdef}(d = 1 - \cos(2a)\ \&\ c = \sin(2a)\ \&\ \neg(\sin a = 0)\ \&\ \neg(\cos a = 0) \to e(\operatorname{tg} a)^b + f = ed^b/c^b + f)$
Они применяются, если усматривается возможность сокращения: либо множитель e делится на d, либо дробь из внешнего контекста сокращается на c. Исключение составляют случаи, когда f является дробью, числитель которой делится на степень синуса a. Если преобразуемое

выражение содержит неизвестные, то a тоже должно их содержать. Уровень срабатывания равен 1.

v. Выражение котангенса через синус и косинус.
$\forall_{abcd}(b-\text{rational}\ \&\ \neg(\text{знаменатель}(b)-\text{even})\ \&\ \neg(\sin d=0)\rightarrow a(\operatorname{ctg} d)^b+c=a(\cos d)^b/(\sin d)^b+c)$
$\forall_{abcdef}(c=1-\cos(2a)\ \&\ d=\sin(2a)\ \&\ \neg(\sin a=0)\rightarrow e(\operatorname{ctg} a)^b+f=ed^b/c^b+f)$
$\forall_{abcdef}(d=1+\cos(2a)\ \&\ c=\sin(2a)\ \&\ \neg(\sin a=0)\ \&\ \neg(\cos a=0)\rightarrow e(\operatorname{ctg} a)^b+f=ed^b/c^b+f)$
Аналогично предыдущему.

vi. Выражение секанса через косинус.
$\forall_{abcd}(b-\text{rational}\ \&\ \neg(\text{знаменатель}(b)-\text{even})\ \&\ \neg(\cos d=0)\rightarrow a(\sec d)^b+c=a/(\cos d)^b+c)$
Преобразуемое выражение либо его внешний контекст должны содержать тригонометрическую операцию, отличную от $\sec d, \operatorname{cosec} d$. Уровень срабатывания равен 2.

vii. Выражение косеканса через синус.
$\forall_{abcd}(b-\text{rational}\ \&\ \neg(\text{знаменатель}(b)-\text{even})\ \&\ \neg(\sin d=0)\rightarrow a(\operatorname{cosec} d)^b+c=a/(\sin d)^b+c)$
Аналогично предыдущему.

viii. Сумма квадратов либо четвертых степеней синуса и косинуса.
После цепочки несложных обобщений упрощающее тождество $k(\sin x)^2+k(\cos x)^2=k$ приобретает следующий вид:
$\forall_{abcdefghijxl}(a-b=2\ \&\ c-d=2\ \&\ (0<e\ \&\ 0<f\ \&\ g=\min(e,f)\ \vee\ e<0\ \&\ f<0\ \&\ g=\max(e,f))\ \&\ h=e-g\ \&\ i=f-g\rightarrow ej(\cos x)^c(\sin x)^b+fj(\cos x)^d(\sin x)^a+l=hj(\cos x)^c(\sin x)^b+ij(\cos x)^d(\sin x)^a+gj(\cos x)^d(\sin x)^b+l)$
Обобщения сводятся к следующему:

   A. Обе части тождества домножаются на произведение произвольных степеней синуса и косинуса x, в результате чего квадраты синуса и косинуса оказываются замаскированы. Чтобы усмотреть их, служат первые два антецедента.

   B. Допускается различие коэффициентов при квадратах синуса и косинуса. При этом они должны представлять собой произведения общей части j на численные множители e, f. Берется тот численный множитель g, который меньше по модулю, из исходных слагаемых выделяются части, соответствующие "общему" коэффициенту g, и их сумма упрощается. Третий антецедент определяет коэффициент g; четвертый и пятый - вычитают его из e, f. Таким образом находятся "остаточные" коэффициенты h, i двух исходных слагаемых.

Степени a, b, c, d идентифицируются с натуральными константами. Указатели "подстановка(фикс(0 1 1 4)x2 0)", "подстановка(фикс(0 1 2 3)x4 0)" разрешают отсутствие множителей $(\sin x)^b$, $(\cos x)^d$. Указатель "пересечениесписков(фикс(0 1 1)фикс(0 1 2))" говорит компилятору, что заголовки первых двух слагаемых могут отличаться от символа "умножение". Указатели "перечень(x5 десчисло(x5))", "перечень(x6 десчисло(x6))" фиксируют способ идентификации множителей e, f, составляя их из всех численных констант соответствующих произведений. Антецеденты выполняют арифметические вычисления и выделены указателем "программа".

По аналогии с только что разобранным тождеством, созданы следующие обобщения тождеств $k - k(\cos x)^2 = k(\sin x)^2$, $k - k(\sin x)^2 = k(\cos x)^2$:
$\forall_{xbcdefghij}(f - e = 2 \mathbin{\&} (0 < c \mathbin{\&} d < 0 \mathbin{\&} g = \min(c, -d) \vee c < 0 \mathbin{\&} 0 < d \mathbin{\&} g = \max(c, -d)) \mathbin{\&} h = c - g \mathbin{\&} i = d + g \to cb(\cos x)^e + db(\cos x)^f + j = hb(\cos x)^e + ib(\cos x)^f + gb(\sin x)^2(\cos x)^e + j)$
$\forall_{xbcdefghij}(d - c = 2 \mathbin{\&} (0 < e \mathbin{\&} f < 0 \mathbin{\&} g = \min(e, -f) \vee e < 0 \mathbin{\&} 0 < f \mathbin{\&} g = \max(e, -f)) \mathbin{\&} h = e - g \mathbin{\&} i = f + g \to eb(\cos x)^c + fb(\cos x)^d + j = hb(\sin x)^c + ib(\sin x)^d + gb(\cos x)^2(\sin x)^c + j)$
Чтобы эти тождества были применены, требуется, чтобы либо преобразуемое выражение имело тригонометрическюу операцию, отличную от рассматриваемой ($\cos x$ в первом приеме и $\sin x$ во втором), либо остаточные коэффициенты h, i и остаточная сумма j равнялись нулю. Аргумент x должен быть неконстантным. Уровень срабатывания равен 2.
Наконец, используется следующее упрощающее тождество для суммы четвертых степеней синуса и косинуса:
$\forall_{abcdef}(b = c \mathbin{\&} d - c = 4 \to a(\sin e)^d(\cos e)^c + a(\sin e)^b(\cos e)^d + f = a(\sin e)^c(\cos e)^c - 2a(\sin e)^{c+2}(\cos e)^{c+2} + f)$
Уровень срабатывания равен 1.

ix. Сумма либо разность тангенсов.

$$\forall_{abcde}\left(a = b + c \to d\operatorname{tg} b + d\operatorname{tg} c + e = \frac{d\sin a}{\cos b\cos c} + e\right)$$

$$\forall_{abcde}\left(a = b - c \to d\operatorname{tg} b - d\operatorname{tg} c + e = \frac{d\sin a}{\cos b\cos c} + e\right)$$

Либо аргументы b, c содержат неизвестные, либо в контексте встречается тригонометрический аргумент, часть слагаемых которого образует выражение a. Уровень срабатывания равен 1.

x. Сумма котангенсов.

$$\forall_{abcde}\left(a = b + c \to d\operatorname{ctg} b + d\operatorname{ctg} c + e = \frac{d\sin a}{\sin b\sin c} + e\right)$$

Аналогично предыдущему.

xi. Разность тангенса и котангенса.
$\forall_{abc}(\neg(\sin b = 0) \mathbin{\&} \neg(\cos b = 0) \to a\operatorname{tg} b - a\operatorname{ctg} b + c = -2a\operatorname{ctg}(2b) + c)$
Указатель "вывод(не(равно(синус(умножение(2 x2))0))эквивалентно(1 2))" обеспечивает регистрацию в списке посылок необходимого для сопровождения по о.д.з. утверждения $\neg(\sin(2b) = 0)$. Оно снабжается комментарием, что первые два антецедента являются его следствиями. Уровень срабатывания равен 1.

xii. Сумма тангенса и котангенса.
$\forall_{abc}(\neg(\sin b = 0) \mathbin{\&} \neg(\cos b = 0) \to a\operatorname{tg} b + a\operatorname{ctg} b + c = 2a\sin(2b) + c)$

Если c имеет слагаемое вида $p\operatorname{tg} b(\sin b)^2$, то прием блокируется, так как в этом случае предпочтительнее выразить сумму через $\operatorname{tg} b$. Уровень срабатывания равен 1.

xiii. Сумма квадрата тангенса либо котангенса и единицы.

$$\forall_{abc}\left(a + a(\operatorname{tg} b)^2 + c = \frac{a}{(\cos b)^2} + c\right)$$

$$\forall_{abc}\left(a + a(\operatorname{ctg} b)^2 + c = \frac{a}{(\sin b)^2} + c\right)$$

Если b содержит неизвестные, причем преобразуемое выражение не содержит тригонометрических операций, отличных от $\operatorname{tg} b, \operatorname{ctg} b$, то преобразование блокируется. Уровень срабатывания равен 1.

xiv. Разность квадратов тангенса и котангенса.
$\forall_{abc}(\neg(\sin b = 0)\ \&\ \neg(\cos b = 0) \to a(\operatorname{ctg} b)^2 - a(\operatorname{tg} b)^2 + c =$
$4a\cos(2b)/(\sin(2b))^2 + c)$

Уровни срабатывания этого и двух следующих приемов равны 1.

xv. Сумма квадратов тангенса и котангенса.
$\forall_{abc}(\neg(\sin b = 0)\ \&\ \neg(\cos b = 0) \to a(\operatorname{tg} b)^2 + a(\operatorname{ctg} b)^2 + c = 4a/(\sin(2b))^2 -$
$2a + c)$

xvi. Сумма кубов тангенса и котангенса.
$\forall_{abc}(\neg(\sin b = 0)\ \&\ \neg(\cos b = 0) \to a(\operatorname{tg} b)^3 + a(\operatorname{ctg} b)^3 + c = 8a/(\sin(2b))^3 -$
$6a/\sin(2b) + c)$

xvii. Синус либо косинус суммы.

Формулы синуса либо косинуса суммы используются, если они дают новое слагаемое, подобное уже имевшемуся:
$\forall_{abcdef}(c = \sin a\cos b\ \&\ d = \cos a\sin b \to f + e\sin(a + b) = f + ec + ed)$
$\forall_{abcdef}(c = \cos a\cos b\ \&\ d = \sin a\sin b \to f + e\cos(a + b) = f + ec - ed)$
Требуется, чтобы остаточная сумма f содержала слагаемое, делящееся на c либо на d. Учитывается, что c, d могут иметь вид дроби в результате вычисления синуса или косинуса. Дополнительно проверяется, что e не имеет своим сомножителем тригонометрическую операцию от суммы аргументов и что f не имеет слагаемых вида $\pm e\sin(\pm a + x), \pm e\sin(\pm b + x)$ (во втором приеме - $\pm e\cos(\pm a + x), \pm e\cos(\pm b + x)$). Уровень срабатывания равен 2.

Для случая суммы с $\pi/4$ добавлены следующие приемы:
$\forall_a(\sin(a + \pi/4) = (\sin a + \cos a)/\sqrt{2})$
$\forall_a(\cos(a + \pi/4) = (\cos a - \sin a)/\sqrt{2})$
Они срабатывают на уровне 3, если приводят к возможности сокращения внешней дроби либо вынесению общего множителя числителей или знаменателей двух дробей.

Наконец, имеется прием, учитывающий возможность вынесения за скобку общего множителя $\cos a - \sin a$ двух слагаемых $b\cos(a + \pi/4), k\cos(2a)$:
$\forall_{abcd}(d = b(\cos a - \sin a)/\sqrt{2} + c \to b\cos(a + \pi/4) + c = d)$
Остаточная сумма c должна иметь слагаемое вида $k\cos(2a)$. Антецедент предпринимает попытку разложить d на множители, обращаясь к нормализатору "видумножение". Числитель результата должен иметь либо два сомножителя с неизвестными, либо степенной сомножитель с неизвестным основанием. Уровень срабатывания равен 3.

xviii. Тангенс суммы. Формула тангенса суммы применяется в двух случаях: если слагаемое a содержит неизвестные, а $\operatorname{tg} a$ уже встречается в преобразуемом выражении, либо если a имеет вид арктангенса.

$\forall_{abcde}(\neg(\cos a = 0) \mathbin{\&} \neg(\cos b = 0) \mathbin{\&} \neg(\cos(a+b) = 0) \rightarrow c\operatorname{tg}(a+b)/d+e = c(\operatorname{tg} a + \operatorname{tg} b)/d(1 - \operatorname{tg} a \operatorname{tg} b) + e)$

Для указанных случаев созданы два отдельных приема. В первом случае дополнительно требуется отсутствие в c, d, e неизвестной тригонометрической операции, отличной от $\operatorname{tg} a$. Если $d = 1$, $e = \pm 1$, а выражение c имеет заголовок "тангенс"или "котангенс", то прием блокируется. Во втором случае единственное условие срабатывания - заголовок "арктангенс" выражения a. Так как первый антецедент тогда автоматически истинен, то он отброшен. Уровень срабатывания приемов равен 1.

xix. Усмотрение тангенса либо котангенса в дроби с синусами и косинусами.

$\forall_{abcdefg}(f = e/2-b/2+\pi/4 \mathbin{\&} \neg(\cos e+\sin b = 0) \mathbin{\&} c-\mathrm{rational} \mathbin{\&} \neg(\text{знаменатель}(c) - \mathrm{even}) \rightarrow a(\cos b+\sin e)^c/d(\cos e+\sin b)^c+g = a(\operatorname{tg} f)^c/d+g)$

$\forall_{abcdefg}(f = e/2+b/2+\pi/4 \mathbin{\&} \neg(\cos e-\sin b = 0) \mathbin{\&} c-\mathrm{rational} \mathbin{\&} \neg(\text{знаменатель}(c) - \mathrm{even}) \rightarrow a(\cos b+\sin e)^c/d(\cos e-\sin b)^c+g = a(\operatorname{tg} f)^c/d+g)$

В обоих случаях выражение g не имеет тригонометрического аргумента, отличного от f.

$\forall_{abcdefg}(b = a/2 \mathbin{\&} \neg(\cos b = 0) \mathbin{\&} \neg(f - f\sin a = 0) \rightarrow c(1+\sin a)^d/e(f - f\sin a)^d + g = c(1+\operatorname{tg} b)^{2d}/ef^d(1-\operatorname{tg} b)^{2d} + g)$

Выражение $\operatorname{tg} b$ должно встречаться в рассматриваемом выражении еще до преобразования. При этом c, e, g не должны иметь тригонометрических операций, отличных от $\operatorname{tg} b, \operatorname{ctg} b$. Уровень срабатывания приемов равен 1.

xx. Сокращение на сумму либо разность синуса и косинуса. Приемы используют переход к половинному аргументу для последующего сокращения дроби:

$\forall_{abcdefgh}(a = \min(b,c) \mathbin{\&} \neg(\cos e = 0) \rightarrow d(h - h\sin e)^b/f(\cos e)^c + g = d(h-h\sin e)^{b-a}h^a(\cos(e/2)-\sin(e/2))^a/f(\cos e)^{c-a}(\cos(e/2)+\sin(e/2))^a + g)$

$\forall_{abcdefg}(a = \min(b,c) \mathbin{\&} \neg(\cos e = 0) \rightarrow d(1 + \sin e)^b/f(\cos e)^c + g = d(1+\sin e)^{b-a}(\cos(e/2)+\sin(e/2))^a/f(\cos e)^{c-a}(\cos(e/2)-\sin(e/2))^a + g)$

$\forall_{abcdefg}(a = \min(b,c) \mathbin{\&} \neg(1 + \sin e = 0) \rightarrow f(\cos e)^c/d(1+\sin e)^b + g = f(\cos e)^{c-a}(\cos(e/2)-\sin(e/2))^a/d(1+\sin e)^{b-a}(\cos(e/2)+\sin(e/2))^a + g)$

$\forall_{abcdefgh}(a = \min(b,c) \mathbin{\&} \neg(h-h\sin e = 0) \rightarrow f(\cos e)^c/d(h-h\sin e)^b+g = f(\cos e)^{c-a}(\cos(e/2)+\sin(e/2))^a/d(h-h\sin e)^{b-a}h^a(\cos(e/2)-\sin(e/2))^a + g)$

Переменные b, c идентифицируются с натуральными константами. Преобразуемое выражение должно иметь тригонометрический аргумент, целым кратным которому является половинный аргумент $e/2$. Уровень срабатывания приемов равен 1.

xxi. Сокращение на косинус половинного аргумента.

$\forall_{abcdefg}(f = \min(b,c) \ \& \ \neg(\sin a = 0) \rightarrow d(1+\cos a)^b/e(\sin a)^c + g = d(1+\cos a)^{b-f}(\cos(a/2))^f/e(\sin a)^{c-f}(\sin(a/2))^f + g)$

$\forall_{abcdefg}(f = \min(b,c) \ \& \ \neg(1+\cos a = 0) \rightarrow e(\sin a)^c/d(1+\cos a)^b + g = e(\sin a)^{c-f}(\sin(a/2))^f/d(1+\cos a)^{b-f}(\cos(a/2))^f + g)$

Переменные b, c идентифицируются с натуральными константами. Либо $b = c$ и тригонометрический аргумент a не встречается вне преобразуемой дроби, либо рассматриваемое выражение имеет тригонометрический аргумент, целым кратным которому служит $a/2$. Уровень срабатывания равен 1.

xxii. Разность четвертых степеней синуса и косинуса.
$\forall_{ab}(a(\cos b)^4 - a(\sin b)^4 = a(\cos b)^2 - a(\sin b)^2)$
Преобразование применяется без ограничений. Уровень срабатывания равен 1.

xxiii. Выражение частного одинаковых степеней синуса и косинуса через тангенс.

$$\forall_{abcde}\left(\frac{a(\sin b)^c}{d(\cos b)^c} + e = \frac{a(\operatorname{tg} b)^c}{d} + e\right)$$

Тригонометрический аргумент b содержит неизвестные. Преобразуемое выражение имеет вхождение операции $\operatorname{tg} b$ либо $\operatorname{ctg} b$; выражения a, d, e не имеют тригонометрических операций, отличных от двух последних. Уровень срабатывания равен 1.

xxiv. Усмотрение синуса либо косинуса суммы либо разности.
$\forall_{abcde}(e = b - c \rightarrow a\sin b\cos c - a\sin c\cos b + d = a\sin e + d)$
$\forall_{abcde}(e = b + c \rightarrow a\sin b\cos c + a\sin c\cos b + d = a\sin e + d)$
$\forall_{abcde}(e = b - c \rightarrow a\cos b\cos c + a\sin c\sin b + d = a\cos e + d)$
$\forall_{abcde}(e = b + c \rightarrow a\cos b\cos c - a\sin c\sin b + d = a\cos e + d)$
Преобразование применяется, если при сложении либо вычитании исходных аргументов b, c происходят сокращения и результат e не имеет заголовка "плюс". Дополнительно проверяется, что либо один из аргументов b, c неконстантный, либо ни один из них не содержит обратных тригонометрических функций. Уровень срабатывания равен 1.

xxv. Переход к тангенсу.
Если преобразуемое выражение имеет вид дроби, то переход от частного синуса и косинуса к тангенсу может выполняться для последующих упрощений внешнего контекста.

$$\forall_{abcd}\left(\frac{a(\sin b)^c}{d(\cos b)^c} = \frac{a(\operatorname{tg} b)^c}{d}\right)$$

Условием применения приема является либо возможность последующего сокращения на $\operatorname{tg} b$ дроби внешнего контекста, либо возможность вынесения этого тангенса за скобку из числителей или знаменателей двух дробей внешнего контекста. Уровень срабатывания равен 4.

xxvi. Усмотрение квадрата синуса либо косинуса в произведении двух сумм с единицами. Для извлечения квадратного корня из $(1-\cos b)(1+\cos b)$ введен следующий прием:
$\forall_{abcd}(a(1-\cos b)^c(1+\cos b)^c + d = a((\sin b)^2)^c + d)$

Показатель степени c идентифицируется с константной дробью, имеющей четный знаменатель. Уровень срабатывания равен 1.

xxvii. Усмотрение возможности сокращения в сумме двух произведений тригонометрических операций. Приемы усматривают возможность сокращения после преобразования двух таких произведений в суммы:
$\forall_{abcde}(a+b-d-e=0 \to c\sin a\sin b-c\sin d\sin e=(c\cos(a-b)-c\cos(d-e))/2)$
$\forall_{abcde}(a+b-d-e=0 \to c\cos a\cos b-c\cos d\cos e=(c\cos(a-b)-c\cos(d-e))/2)$
$\forall_{abcde}(a+b-d-e=0 \to c\sin a\sin b+c\cos d\cos e=(c\cos(a-b)+c\cos(d-e))/2)$
$\forall_{abcde}(a+b-d-e=0 \to c\sin a\cos b-c\sin d\cos e=(c\sin(a-b)-c\sin(d-e))/2)$
Антецедент обрабатывает левую часть равенства нормализаторами "нормплюс", "стандплюс". Уровень срабатывания равен 3.

xxviii. Сумма косинуса двойного аргумента и квадрата косинуса либо синуса. Происходит преобразование квадрата в косинус двойного аргумента и приведение подобных членов:
$\forall_{abcde}(d=2c \to a\cos d+b(\cos c)^2+e=(a+b/2)\cos d+b/2+e)$
$\forall_{abcde}(d=2c \to a\cos d+b(\sin c)^2+e=(a-b/2)\cos d+b/2+e)$
Аргумент c одержит неизвестные; коэффциенты a,b - не содержат. Уровень срабатывания равен 2.

xxix. Усмотрение возможности сокращения при возведении в квадрат суммы либо разности синуса и косинуса.
$\forall_{abcd}(a-a(b\sin d+c\cos d)^2=-2abc\sin d\cos d)$
Коэффициенты b,c идентифицируются с ±1. Уровень срабатывания равен 2.

(d) Попытки перехода к новому тригонометрическому аргументу. В основном, приемы данного раздела ориентированы на уменьшение числа различных тригонометрических аргументов в выражении.

i. Переход к половинному аргументу.

При наличии комментария "уравнменьше" (попытка разложения на множители разности частей неравенства) применяются следующие приемы, позволяющие получить выражение с единственной тригонометрической операцией - синусом либо косинусом половинного аргумента:
$\forall_{abcd}(d=b/2 \to a\cos b+c=2a(\cos d)^2-a+c)$
$\forall_{abcd}(d=b/2 \to a\cos b+c=a-2a(\sin d)^2+c)$
В первом случае преобразуемое выражение должно иметь вхождение $\cos d$ и не иметь других тригонометрических операций, кроме $\cos b$, $\cos d$, $\sin d$, причем последние - только в четных степенях. Второй случай - аналогичен, с заменой $\cos d$ на $\sin d$ и обратно. Остаточная сумма c должна быть неконстантной. Уровень срабатывания равен 1.

На уровне 5 предпринимаются попытки перехода к половинному аргументу с рекурсивным обращением к нормализатору "видумножение":
$\forall_{abcde}(d=\cos(b/2)\ \&\ e=2ad^2-a+c \to a\cos b+c=e)$
$\forall_{abcde}(d=\sin(b/2)\ \&\ e=a-2ad^2+c \to a\cos b+c=e)$
Здесь первый антецедент упрощает тригонометрическую операцию d от половинного аргумента с помощью нормализаторов общей стандар-

тизации, второй - обращается к нормализатору "видумножение". Замена реализуется лишь при получении выражения e мультипликативного вида. Для инициализации попытки необходимо, чтобы остаточная сумма c имела слагаемое, делящееся на d, причем преобразуемое выражение не содержало синусов и косинусов, отличных от $\cos b$, d. Остаточная сумма c должна быть неконстантной.

В случае синуса переход к половинному аргументу выполняется следующим приемом:

$\forall_{abcdefn}(d = \sin(b/2)\ \&\ e = \cos(b/2)\ \&\ f = a2^n d^n e^n + c \rightarrow a(\sin b)^n + c = f)$

Первый и второй антецеденты упрощают синус и косинус половинного аргумента, третий - обращается к нормализатору "видумножение". Проверяется, что результат f имеет мультипликативный вид. Для инициализации попытки необходимо, чтобы каждое слагаемое остаточной суммы c делилось на d либо на e. Уровень срабатывания равен 5.

ii. Переход к аргументу, деленному на 3.

$\forall_{abcde}(d = \cos(b/3)\ \&\ e = 4ad^3 - 3ad + c \rightarrow a\cos b + c = e)$

$\forall_{abcde}(d = \sin(b/3)\ \&\ e = -4ad^3 + 3ad + c \rightarrow a\sin b + c = e)$

Первый антецедент упрощает новую тригонометрическую операцию d, второй - обращается к нормализатору "видумножение". Проверяется, что результат e имеет мультипликативный вид. Для инициализации попытки необходимо выполнение следующих требований:

A. Каждое неконстантное слагаемое остаточной суммы c, не делящееся на d, должно делиться на $\cos(2b/3)$ (у второго приема - на $\sin(2b/3)$) и содержать неизвестные.

B. Либо a не имеет тригонометрических операций, либо преобразуемое выражение не имеет тригонометрических аргументов, отличных от b и того аргумента, который после упрощений получается в d.

C. Каждое неконстантное слагаемое выражения c имеет общий параметр с b.

D. Каждая неизвестная, входящая в c, встречается в b.

E. Остаточная сумма c неконстантная.

Уровень срабатывания приемов равен 5.

Если тригонометрический аргумент b имеет вид суммы, то для усиления его сравнения с аргументом, деленным на 3, созданы два дополнительных приема:

$\forall_{abcdempq}(p\sin b = \sin(3d)\ \&\ q = (-4a(\sin d)^3 + 3a\sin d)/p + c(\sin d)^m + e \rightarrow a\sin b + c(\sin d)^m + e = q)$

$\forall_{abcdempq}(p\cos b = \cos(3d)\ \&\ q = (4a(\cos d)^3 - 3a\cos d)/p + c(\cos d)^m + e \rightarrow a\cos b + c(\cos d)^m + e = q)$

Аргумент d имеет заголовок "плюс". Первый антецедент упрощает $\sin(3d)$ и убеждается, что результат делится на $\sin b$ (p равно ± 1). Второй антецедент обращается к нормализатору "видумножение". Первый прием применяется, если все тригонометрические операции преобразуемого выражения суть синусы, второй - косинусы. Уровень срабатывания равен 6.

iii. Переход к двойному аргументу.

A. Четная степень синуса.

$\forall_{abcde}(a = 2b \to d(\sin c)^a + e = d(1 - \cos(2c))^b/2^b + e)$
Показатель степени a идентифицируется с натуральной константой. Для применения приема необходимо выполнение следующих требований:

Преобразуемое выражение не имеет отличного от c тригонометрического аргумента, кратным которого служит c.

Преобразуемое выражение не имеет тригонометрического аргумента, отличающегося от c, быть может, лишь заменой знака и добавлением рациональных кратных пи, который встречался бы в контексте, не допускающем перехода к двойному аргументу. В частности, каждое вхождение c должно допускать такой переход. Это и предыдущее требования встречаются в нескольких различных приемах. Поэтому они сгруппированы в отдельный оператор фильтра "фильтрудвоения", который учитывает также ряд дополнительных редко встречающихся особенностей. Оператор реализован по дизъюнктивному принципу - он перечисляет отрицания условий срабатывания, так что соответствующий фильтр приема имеет вид "не(фильтрудвоения(x1))".

Коэффициент d не имеет своим сомножителем степень косинуса c.

Выражение $d + e$ после вынесения за скобку общих множителей слагаемых d, e, не становится разностью квадратов.

Выражение e не имеет сомножителем своего числителя степень синуса.

Прием срабатывает на уровне 2.

B. Четная степень косинуса.
$\forall_{abcde}(a = 2b \to d(\cos c)^a + e = d(1 + \cos(2c))^b/2^b + e)$
Аналогично предыдущему.

C. Произведение синуса на косинус.
$\forall_{abcd}(c(\sin a)^b(\cos a)^b + d = c(\sin(2a))^b/2^b + d)$
Либо истинно "не(фильтрудвоения(x1))", либо внешний контекст уже имеет вхождение тригонометрического аргумента $2a$, причем каждое вхождение аргумента a в преобразуемое выражение допускает переход к $2a$. Уровень срабатывания равен 2.

D. Произведение суммы тангенса и единицы на их разность.
$\forall_{abcdefg}(a = \cos(2b)\ \&\ \neg(\cos b = 0)\ \&\ c\text{—rational}\ \&\ \neg(\text{знаменатель}(c)\text{—even}) \to (d - d\,\mathrm{tg}\, b)^c(1 + \mathrm{tg}\, b)^c e/f + g = 2^c d^c a^c e/(a + 1)^c f + g)$
Преобразуемое выражение имеет тригонометрический аргумент, кратный либо равный $2b$. Уровень срабатывания равен 1.

E. Четная степень тангенса.
$\forall_{abcfg}(b = 2c\ \&\ \neg(\cos a = 0) \to f(\mathrm{tg}\, a)^b + g = f((1 - \cos(2a))/(1 + \cos(2a)))^c + g)$
Преобразуемое выражение должно иметь тригонометрический аргумент, отличный от a. Либо истинно "не(фильтрудвоения(x1))", либо показатель степени b отличен от 2. Выражение g не имеет слагаемого, сомножителем числителя которого служит степень косинуса a. Это выражение не имеет сомножителем своего числителя степень синуса a. Уровень срабатывания равен 1.

F. Четная степень котангенса.
$\forall_{abcfg}(b=2c\ \&\ \neg(\sin a=0)\rightarrow f(\mathrm{ctg}\,a)^b+g=f((1+\cos(2a))/(1-\cos(2a)))^c+g)$
Аналогично предыдущему.

G. Усмотрение тангенса двойного аргумента.
$\forall_{abcde}(\neg(\cos b=0)\ \&\ \neg(c-c(\mathrm{tg}\,b)^2=0)\rightarrow a\,\mathrm{tg}\,b/e(c-c(\mathrm{tg}\,b)^2)+d=a\,\mathrm{tg}(2b)/2ce+d)$
Коэффициент c идентифицируется с ненулевой десятичной константой. Уровень срабатывания равен 1.

iv. Переход к тройному аргументу.

A. Усмотрение синуса тройного угла.
$\forall_{abcd}(d=\sin(3a)\rightarrow b\cos(a+\pi/6)\cos(-a+\pi/6)\sin a+c=bd/4+c)$
Преобразование применяется, если остаточная сумма c имеет слагаемое, делящееся на d. Уровень срабатывания равен 2.
$\forall_{abcd}(d=\cos(\pi/6-b/3)\cos(\pi/6+b/3)\sin(b/3)\rightarrow a\sin b+c=4ad+c)$
Хотя бы один из сомножителей произведения d, содержащий тригонометрическую операцию, должен делить слагаемое выражения c. Это выражение должно иметь тригонометрический аргумент с заголовком "плюс". Уровень срабатывания равен 6.

B. Усмотрение косинуса тройного угла.
$\forall_{abcd}(d=\cos(3a)\rightarrow b\sin(a+\pi/6)\sin(-a+\pi/6)\cos a+c=bd/4+c)$
$\forall_{abcd}(d=\sin(\pi/6-b/3)\sin(\pi/6+b/3)\cos(b/3)\rightarrow a\cos b+c=4ad+c)$
Аналогично предыдущему.

C. Усмотрение тангенса тройного аргумента.
$\forall_{abcd}(\neg(\cos b=0)\ \&\ \neg(1-3(\mathrm{tg}\,b)^2=0)\rightarrow a\,\mathrm{tg}\,b(3-(\mathrm{tg}\,b)^2)/c(1-3(\mathrm{tg}\,b)^2)+d=a\,\mathrm{tg}(3b)/c+d)$
Уровень срабатывания равен 1.

v. Произведение тригонометрической операции на тригонометрическую операцию тройного аргумента. Преобразование такого произведения к виду суммы дает два новых тригонометрических аргумента, один из которых вдвое больше другого:
$\forall_{abpqn}(b=\cos(4a)\rightarrow p(\sin a)^n\sin(3a)+q=(p(\sin a)^{n-1}\cos(2a)-p(\sin a)^{n-1}b)/2+q)$
$\forall_{abpqn}(b=\cos(4a)\rightarrow p(\cos a)^n\cos(3a)+q=(p(\cos a)^{n-1}\cos(2a)-p(\cos a)^{n-1}b)/2+q)$
Выражение a содержит неизвестные. Преобразуемое выражение должно содержать тригонометрическую операцию с тем аргументом, который возникает у b после упрощения. Остаточная сумма q не должна иметь слагаемое, делящееся на произведение рассматриваемых тригонометрических операций от a, $3a$. Уровень срабатывания равен 2. В специальных случаях сразу выполняется переход к двойному аргументу:
$\forall_{abpq}(b=\cos(2a)\rightarrow p\cos a\cos(3a)+q=(pb+2pb^2-p)/2+q)$
Аргумент a содержит неизвестные. Выражения p,q не имеют неизвестных тригонометрических операций, отличных от b. Выражение q не имеет вида $-pb$ и не имеет слагаемого вида $-pb/2$. Уровень срабатывания равен 2.

vi. Переход от произведения тригонометрических функций к сумме. Пре-

образование применяется, если вместо двух старых тригонометрических аргументов, не имевших других вхождений в преобразуемое выражение, возникают два новых, уже в нем встречающихся:
$\forall_{abcdef}(c = \sin(a-b)\ \&\ d = \sin(a+b) \to e\sin a\cos b + f = e(c+d)/2 + f)$
$\forall_{abcdef}(c = \cos(a-b)\ \&\ d = \cos(a+b) \to e\cos a\cos b + f = e(c+d)/2 + f)$
$\forall_{abcdef}(c = \cos(a-b)\ \&\ d = \cos(a+b) \to e\sin a\sin b + f = e(c-d)/2 + f)$
Здесь ни один из содержащих неизвестные тригонометрических аргументов a, b не имеет других вхождений в преобразуемое выражние. Каждый содержащий неизвестные тригонометрический аргумент выражений c, d встречается в e либо в f. Уровень срабатывания равен 2.

(e) Попытка перехода к сумме тригонометрических операций. Это - тригонометрический аналог рассмотренного выше приема раскрывания скобок при разложении на множители. Теорема приема имеет вид $\forall_{ab}(b = a \to a = b)$. Выражение a сначала обрабатывается нормализатором "стандплюс", снабженным комментарием "видумножение". Это обеспечивает преобразование тригонометрических произведений в суммы. Затем оно обрабатывается нормализатором "видумножение". Для блокировки повторной попытки используется комментарий "тригонометрия". Указатель "выход" приводит к немедленной выдаче результата после применения приема. Созданы две версии приема, срабатывающие, соответственно, на уровнях 2 и 6. Для реализации первой из них необходимо выполнение следующих требований:

i. Выражение a содержит неизвестные и не имеет мультипликативного заголовка либо заголовка "минус".
ii. Отсутствует комментарий "группировка", свидетельствующий о начале попыток группировки слагаемых.
iii. Выражение a либо имеет произведение с не менее чем двумя операциями $\sin, \cos$, либо имеет натуральную степень синуса или косинуса.
iv. Если a неконстантное, то содержит сумму либо произведение, у которых хотя бы два различных операнда имеют общий параметр.
v. a либо имеет слагаемое с не менее чем двумя вхождениями тригонометрических аргументов - неизвестных сумм, либо имеет более одного слагаемого с более чем одним вхождением неизвестного тригонометрического аргумента, либо имеет слагаемое с более чем двумя неизвестными тригонометрическими аргументами.
vi. a не имеет слагаемого, основание степени сомножителя числителя которого представляло бы собой сумму (в этом случае сначала должен сработать прием "обычного" раскрывания скобок).
vii. a не имеет дробных слагаемых с неконстантным знаменателем.
viii. Число вхождений тригонометрических операций в a более 1.
ix. Выполнены требования, накладываемые оператором "фильтрмножителей", который уже рассматривался выше в приеме решения уравнений путем разложения на множители разности частей. Он оценивал целесообразность использования найденного разложения, сравнивая его с исходным уравнением.

Вторая версия приема проверяет выполнение следующих требований:

i. См. пункты i,ii,iv первой версии;

   ii. a имеет либо произведение не менее чем двух операций $\sin, \cos$, либо натуральную степень синуса или косинуса, не расположенные в основании степени с дробным показателем.
   iii. Если a содержит неизвестные, то имеет не менее двух различных тригонометрических операций.
   iv. Если a содержит неизвестнеые, то числитель результата b не имеет своим сомножителем сумму с более чем двумя вхождениями тригонометрических операций.
   v. a не содержит неконстантных логарифмов.

(f) Специальные преобразования выражений с неизвестными.

   i. Переход к выражению с идентичными тригонометрическими операциями.
   $\forall_{ade}(d(\cos a)^2 + e = d - d(\sin a)^2 + e)$
   a содержит неизвестные. e содержит $\sin a$ и не содержит других неизвестных тригонометрических операций. Коэффициент d не содержит неизвестных; e не имеет вида $\pm k(\sin a)^2$. Уровень срабатывания равен 2. Аналогичный прием имеется для перехода к косинусу:
   $\forall_{ade}(d(\sin a)^2 + e = d - d(\cos a)^2 + e)$
   ii. Попытка перехода к косинусу половинного аргумента. Если выражение имеет вхождения неизвестного косинуса половинного аргумента, а единственная отличная от этого косинуса неизвестная тригонометрическая операция - косинус данного аргумента, то она выражается через половинный аргумент:
   $\forall_{abcdef}(e = \cos(b/2)\ \&\ f = a(2e^2 - 1)^c + d \rightarrow a(\cos b)^c + d = f)$
   Выражение b содержит неизвестные. Первый антецедент упрощает косинус половинного аргумента, а второй - обращается к нормализатору "видумножение". Указатель "выход" обеспечивает немедленную выдачу результата. Числитель выражения f не должен иметь своим сомножителем сумму с более чем двумя неизвестными тригонометрическими операциями. Уровень срабатывания равен 2.
   iii. Преобразование суммы синуса и косинуса к виду синуса.
   $\forall_{abcd}(d = \sin(c + \pi/4) \rightarrow a\sin c + a\cos c + b = \sqrt{2}ad + b)$
   Выражение c содержит неизвестные. В a, b нет тригонометрических аргументов, отличных от того, который возникает в выражении d после упрощения. Уровень срабатывания равен 2.
   iv. Приведение подобных членов с неизвестной тригонометрической операцией.
   $\forall_{abcd}(a\sin b + c\sin b + d = (a + c)\sin b + d)$
   $\forall_{abcd}(a\cos b + c\cos b + d = (a + c)\cos b + d)$
   Коэффициенты a, b не содержат неизвестных; аргумент b - содержит. Каждое неизвестное слагаемое выражения d отличается от двух группируемых слсгаемых только известным коэффициентом. Уровень срабатывания равен 3.

(g) Обрыв разложения на множители для простейших выражений с неизвестными. Если в результате преобразований получается выражение с неизвестными, для которого уже существуют стандартные средства решения уравнений и неравенств с нулевой противоположной частью, то работа нормализатора "видумножение" обрывается.

i. Сумма произведения синуса и косинуса и их суммы.
$\forall_{abcx}(a\sin x\cos x + b\sin x + b\cos x + c = a\sin x\cos x + b\sin x + b\cos x + c)$
Коэффициенты a, b, c известны; x содержит неизвестные. Указатель "выход"обеспечивает немедленную выдачу текущего выражения как результата обращения к процедуре "видумножение". Предварительно оно обрабатывается нормализаторами общей стандартизации. Уровень срабатывания этого и следующих двух приемов равен 1.

ii. Разность либо сумма одинаковых степеней.
$\forall_{abn}(a^n - b^n = a^n - b^n)$
Выражение содержит неизвестную тригонометрическую операцию. n идентифицируется с натуральной константой.

iii. Линейное выражение с синусом и косинусом.
$\forall_{abcd}(a\sin b + c\cos b + d = a\sin b + c\cos b + d)$
Коэффициенты a, c и слагаемое d не содержат неизвестных; аргумент b - содержит.

Завершая список тригонометрических приемов разложения на множители, возвращаемся к корневому списку разделов нормализатора. Здесь остаются лишь небольшие дополнения.

19. Дополнительные приемы усмотрения разности квадратов.

$\forall_{abcdepquv}(a^2 - d = 0\ \&\ c^2 + e = 0\ \&\ p - \mathrm{rational}\ \&\ \neg(\text{знаменатель}(p) - \mathrm{even}) \to u(a\sqrt{(b)} + c)^q(db+e)^p + v = u(a\sqrt{(b)} + c)^{q+p}(a\sqrt{b} + c)^{q+p}(a\sqrt{b} - c)^p + v)$

$\forall_{abcdepquv}(a^2 - d = 0\ \&\ c^2 + e = 0\ \&\ p - \mathrm{rational}\ \&\ \text{знаменатель}(p) - \mathrm{even}\ \&\ 0 \le db + e\ \&\ 0 \le a\sqrt{b} - c \to u(a\sqrt{(b)} + c)^q(db+e)^p + v = u(a\sqrt{(b)} + c)^{q+p}(a\sqrt{b} + c)^{q+p}(a\sqrt{b} - c)^p + v)$

Усматривается, что множитель $db+e$ есть разность квадратов выражений $a\sqrt{b}, c$. Так как степень суммы этих выражений является другим множителем, то разность квадратов представляется в виде произведения суммы на разность. Прием срабатывает на уровне 4.

20. Вынесение за скобку общего множителя двух конечных произведений. Рассматривается случай, когда множители одного конечного произведения включаются в другое:

$$\forall_{abcdef}\left(0 \le e - c \to a\prod_{n=d}^{c} f(n) + b\prod_{n=d}^{e} f(n) = \prod_{n=d}^{c} f(n)\left(a + b\prod_{n=c+1}^{e} f(n)\right)\right)$$

Переменная f выделена указателем "отображение". Уровень срабатывания равен 3.

21. Разложение на множители биквадратного трехчлена при формальном интегрировании. Несложные преобразования позволяют представить исходное выражение в виде произведения двух квадратных трехчленов. Возникающие при этом радикалы не содержат переменной интегрирования и поэтому приемлемы:

$\forall_{abcdefg}(0 < a \,\&\, 0 < c \,\&\, e = \sqrt{a} \,\&\, f = \sqrt{c} \,\&\, g = 2ef - b \,\&\, 0 < g \rightarrow ah^{4d} + bh^{2d} + c = (eh^{2d} - \sqrt{g}h^d + f)(eh^{2d} + \sqrt{g}h^d + f))$

Имеется комментарий (нормИнтеграл h), означающий, что h - переменная интегрирования. Проверяется, что она не входит в коэффициенты. Разрешается отсутствие члена с коэффициентом b. Уровень срабатывания равен 3.

22. Гиперболические функции. В этом разделе имеется всего один прием:

    $\forall_{ab}(a(\operatorname{ch} b)^2 - a = a(\operatorname{sh} b)^2)$

    Рекомендуется самостоятельно обучить решатель разложению на множители выражений с гиперболическими функциями.

23. Сигнум произведения.

    $\forall_{abcd}(a + \mathrm{sg}(bc)d = a + \mathrm{sg}b\mathrm{sg}cd$

    Требуется, чтобы каждое слагаемое выражения a имело множитель вида $\mathrm{sg}(kb)$, где k может обращаться в единицу. Если само преобразуемое выражение имеет вид сигнума произведения, то применяется другой прием:

    $\forall_{ab}(\mathrm{sg}(ab) = \mathrm{sg}a\mathrm{sg}b)$

    Уровень срабатывания обоих приемов равен 1.

24. Модуль произведения.

    $\forall_{abcd}(a + |bc|d = a + |b||c|d)$

    Каждое слагаемое выражения a должно иметь множитель вида $|kb|$. Уровень срабатывания равен 1.

25. Разность под сигнумом. Если это дает лексикографически предшествующее выражение, выполняется перестановка операндов разности и вынесение минуса наружу:

    $\forall_{abcde}(\mathrm{sg}((b - a)c)d + e = e - \mathrm{sg}((a - b)c)d)$

    Проверяется, что выражение a лексикографически предшествует выражению b. Каждое слагаемое выражения e делится либо на $\mathrm{sg}(k(b-a))$, либо на $\mathrm{sg}(k(a-b))$, причем представлены оба типа слагаемых. Уровень срабатывания равен 1.

26. Нормализация степенных сомножителей.

    $\forall_{abcn}((-a)^n b + c = a^n b + c)$

    Здесь n идентифицируется с четной константой.

    $\forall_{abc}((\sqrt{a})^2 b + c = ab + c)$

27. Разложение многочленов с помощью алгоритма Берлекэмпа.

    Этот алгоритм реализован на ГЕНОЛОГе так, что в нем используются не логические структуры данных, а обычные вычислительные (числа, массивы, и т.п.). Поэтому описание данной реализации будет отложено до тех разделов, где речь пойдет о вычислениях на ГЕНОЛОГе.

Упрощенный нормализатор разложения на множители "факторизация"

Этот нормализатор используется в тех случаях, когда достаточно ограничиться лишь наиболее явно усматриваемыми случаями разложения на множители. Он содержит следующие приемы, дублирующие приемы нормализатора "видумножение":

1. Вынесение за скобку общего множителя всех слагаемых.
2. Произведение двух сумм, имеющих каждая два слагаемых.
3. Разность квадратов.
4. Разность кубов.
5. Сумма кубов.
6. Сумма разности квадратов двух чисел и их же разности.
7. Сумма разности квадратов двух чисел и их же суммы.
8. Куб суммы.
9. Куб разности.
10. Разложение на множители выражения под корневым минусом.
11. Разложение на множители суммы, являющейся сомножителем слагаемого.
12. Сигнум произведения.
13. Перестановка операндов разности под сигнумом.
14. Вынесение за скобку общего множителя двух модулей.

Нормализатор стандартизации произведений с неизвестными "уравнумножение"

1. Группировка под общий неизвестный показатель степеней с известными основаниями.

   $\forall_{abc}(0 \le a \;\&\; 0 \le b \to a^c b^c = (ab)^c)$

   $\forall_{abc}(c - \mathrm{rational} \;\&\; \neg(\text{знаменатель}(c) - \mathrm{even}) \to a^c b^c = (ab)^c)$

   Выражения a, b не содержат неизвестных, а c - содержит. Уровень срабатывания равен 1.

2. Обращение к стандартизации сомножителей.

   $\forall_{abc}(c = a \to ab = bc)$

   Антецедент использует процедуру "нормуравн", которая играет роль диспетчера при обращениях к нормализаторам стандартизации выражений с неизвестными. Уровень обращения равен 2.

3. Умножение на дробь.

   $\forall_{abc}(c(a/b) = (ac/b))$

   Уровень обращения равен 1.

4. Устранение вложенных умножений.

Нормализаторы "предвумножение" и "стандумножение"

Эти нормализаторы используются при стандартизации процедурой "видумножение" числителей и знаменателей дробей. Примеры их приемов приводились выше.

Анализатор "неизвмножители"

Анализатор используется для получения следствий, возникающих при делении друг на друга двух уравнений. Предполагается, что такое деление приводит к сокращению левых частей на неизвестный множитель. В анализаторе имеется всего три приема, один из которых реализует деление уравнений, а два других - выполняют предварительные преобразования, приводящие к срабатыванию первого.

1. Попытка разложения на множители неизвестной суммы - основания степени сомножителя левой части уравнения. Прием "измельчает" неизвестные множители левой части, чтобы при последующем делении уравнений выявились возможные сокращения.

   $\forall_{abcdef}(a(b+c)^f = d \;\&\; e = b+c \to ae^f = d)$

   Первый антецедент идентифицируется с некоторой посылкой внутренней задачи анализатора. Требуется, чтобы основание степени $b+c$ содержало неизвестные. Второй антецедент обращается к нормализатору "видумножение" для разложения данного основания на множители. Проверяется, что результат разложения либо имеет не менее двух неизвестных сомножителей, либо имеет сомножитель - неизвестную степень. Указатель "прообраз(1)" регистрирует в комментариях информацию о том, что исходное уравнение эквивалентно выведенному новому уравнению $ae^f = d$. Уровень срабатывания равен 1.

2. Вывод линейной комбинации уравнений, левые части которых имеют общий неизвестный множитель. Фактически, этот прием выполняет деление уравнений.

   $\forall_{abcde}(ad = b \;\&\; ae = c \to be - cd = 0)$

   Первые два антецедента идентифицируются с уравнениями из списка посылок внутренней задачи анализатора. a - общая часть сомножителей левых частей. Проверяется, что она содержит неизвестные, причем хотя бы одно из остаточных произведений d, e тоже содержит неизвестные. Правые части b, c должны быть известными. Прием выводит линейную комбинацию уравнений (первое домножается на e, второе - на d, и выполняется вычитание), которая может рассматриваться как результат деления уравнений и исключения знаменателей. Ненулевая часть этой линейной комбинации обрабатывается нормализатором "видумножение", а сама линейная комбинация - нормализатором числовых равенств "нормчисло". Если в результате получается дизъюнктивное утверждение либо линейное уравнение, то работа анализатора обрывается, а найденное утверждение пересылается во внешнюю задачу. Это обеспечивается указателем "обрыв(или(заголовок(результат или)линейноеуравнение(результат неизвестные)))". Если уравнения имеют неизвестные тригонометрические операции, то вводится сильное ограничение трудоемкости. Прием срабатывает на уровне 1.

3. Линейная комбинация двух уравнений с целью получить в левой части произведение неизвестных выражений. Прием уничтожает в двух исходных уравнениях

по одному неизвестному слагаемому и пытается разложить полученную новую левую часть на множители:

$\forall_{abcdefghi}(ah + b = c \,\&\, ai + d = e \,\&\, f = bi - dh \,\&\, g = ci - eh \rightarrow f = g)$

Первый и второй антецеденты идентифицируются с посылками анализатора. Их левые части имеют "подобные" слагаемые ah, ai (a содержит неизвестные; h, i - не содержат). Третий антецедент строит линейную комбинацию левых частей, уничтожающую эти подобные слагаемые, и обращается к нормализатору "видумножение". Четвертый антецедент строит соответствующую линейную комбинацию правых частей и раскрывает в ней скобки с помощью нормализатора "стандплюс". Левые части уравнений должны быть свободны от дробных слагаемых. Проверяется, что новая левая часть f либо имеет не менее двух неизвестных сомножителей, либо имеет сомножителем неизвестную степень. Результирующее уравнение обрабатывается нормализатором "нормчисло". Если при этом возникает дизъюнкция нескольких линейных уравнений, то работа анализатора обрывается, а дизъюнкция передается во внешнюю задачу. Уровень срабатывания равен 2.

10.7 Приемы символа "дробь"

Выражение "дробь(a b)", обозначающее частное от деления a на b, прорисовывается формульным редактором в виде

$$\frac{a}{b}\,.$$

Двоеточие и косая черта не предусмотрены. Справочники "арность", "одз", "типданных", "область", "единица", "заменазнака" определяют простейшие свойства операции (имеет два операнда, определена на вещественных числах, знаменатель должен быть отличен от нуля, операция принимает любые вещественные значения и имеет единицу 1 для второго операнда, допускает вынесение минуса из знаменателя и числителя). Справочник "монотонно" указывает, что значение дроби, имеющей положительный знаменатель, монотонно неубывает при монотонном неубывании числителя. Справочник "конст" указывает, что при нулевом числителе значение дроби обращается в ноль. Справочник "вычисл" определяет операторы ЛОСа, используемые для деления чисел, представленных в вычислительных форматах "с плавающей запятой" и "длинное целое". Ряд других справочников предоставляет для символа "дробь" информацию, используемую процедурами базы теорем и генератора приемов.

Общая стандартизация выражений

Для простейшей стандартизации дробных выражений созданы следующие приемы:

1. Ноль в числителе.

$$\forall_a \left(\frac{0}{a} = 0\right)$$

Уровень срабатывания равен 0. Указатель "нормализатор" инициирует обращение к процедуре дополнительной стандартизации надвыражения после выполнения замены. Таким образом уменьшается число сканирований.

2. Единица в знаменателе.

$$\forall_a \left(\frac{a}{1} = a\right)$$

Аналогично предыдущему.

3. Сокращение числовой дроби.

$$\forall_{abc} \left(c = \frac{a}{b} \to \frac{a}{b} = c\right)$$

Переменные a, b идентифицируются с десятичными константами. Антецедент обращается к нормализатору "сокращдроби", реализованному на ЛОСе и обеспечивающему получение терма - простой несократимой дроби, быть может, расположенной под операцией "минус". Числитель этой дроби неотрицателен, знаменатель положителен. Если a равно 1 либо c совпадает с исходной дробью, применение приема блокируется. Уровень срабатывания равен 0.

4. Приведение числовой дроби к виду составной дроби.

$$\forall_{abc} \left(c = \frac{a}{b} \to \frac{a}{b} = c\right)$$

Переменные a, b идентифицируются с натуральными константами, причем b меньше a. Антецедент преобразует выражение к виду составной дроби, используя нормализатор "составнаядробь". Этот нормализатор реализован на ЛОСе и обеспечивает получение терма - суммы натурального числа и простой несократимой дроби, числитель которой меньше знаменателя. Переход к составной дроби в середине вычислений редко бывает целесообразен. Поэтому прием применяется всего в двух случаях: если дробь - корневая, т.е. не является частью какого-либо внешнего терма, и если она является слагаемым суммы, уже имеющей в своем составе десятичную константу. Эта константа будет сложена с натуральной частью составной дроби, и вид выражения не усложнится. Уровень срабатывания равен 0.

Несколько дополнительных приемов типа преобразования к составной дроби относятся к конкретным единицам измерения (например, перевод дробного числа рублей в рубли и копейки).

5. Сокращение числовых множителей в числителе и знаменателе дроби.

$$\forall_{abcd} \left(\frac{ad}{bc} = \frac{(a/b)d}{c}\right)$$

Переменные a, b идентифицируются с десятичными константами. Выражение a/b в заменяющем терме обрабатывается нормализатором "сокращдроби". Он реализован на ЛОСе и дает терм для простой несократимой дроби. Проверяется, что дробь удалось сократить. Если преобразуемое дробное выражение уже является дробью с натуральными числителем и знаменателем, то попытка сокращения блокируется. Уровень срабатывания равен 0.

6. Умножение на дробь.

   За исключением особых случаев (умножение дроби на единицу измерения либо умножение на дробную единицу измерения), внешние множители переносятся в числитель дроби:

   $$\forall_{abc}\left(c\frac{a}{b}=\frac{ac}{b}\right)$$

   Уровень срабатывания равен 0.

7. Деление на дробь.

   $$\forall_{abc}\left(\frac{b}{c/a}=\frac{ab}{c}\right)$$

   Прием применяется без ограничений. Уровень срабатывания равен 0.

8. Деление дроби.

   $$\forall_{abc}\left(\frac{b}{ac}=\frac{b/c}{a}\right)$$

   Направление замены - справа налево. Остальное - аналогично предыдущему приему.

9. Вынесение минуса из числителя.

   $$\forall_{ab}\left(-\frac{b}{a}=\frac{-b}{a}\right)$$

   Направление замены - справа налево. Уровень срабатывания равен 0.

10. Вынесение минуса из знаменателя.

    $$\forall_{ab}\left(\frac{a}{-b}=-\frac{a}{b}\right)$$

    Уровень срабатывания равен 0.

11. Сокращение дроби.

    $$\forall_{abc}\left(\frac{ac}{bc}=\frac{a}{b}\right)$$

    Компилятор использует для нахождения общего множителя c процедуру "алгебрпересечение", так что прием усматривает возможность сокращения даже в несколько замаскированных ситуациях. Например, числитель и знаменатель могут быть не произведениями, а степенями с одинаковыми основаниями, натуральными числами и т.п. Отличие множителя c от нуля специально не проверяется, так как вытекает из условий на о.д.з. преобразуемой дроби. Тем не менее, в особых случаях, когда такая проверка может понадобиться (дробь расположена под описателем "отображение", но не внутри интеграла), ее выполняет один из фильтров. Остаточные множители a, b обрабатываются нормализатором "факторизация", а результирующая дробь - нормализатором "нормдробь". Здесь может произойти дополнительное сокращение. Уровень срабатывания равен 1.

Чтобы усматривать возможность сокращения на общий константный множитель a слагаемых числителя, добавлен еще один прием:

$$\forall_{abcde}\left(\neg(a=0)\rightarrow\frac{(ab+ac)e}{ad}=\frac{(b+c)e}{d}\right)$$

Переменная a идентифицируется с константным выражением. Обычно прием срабатывает, если b, c тоже константные. Например, он необходим для замены $(2-\sqrt{2})/2\sqrt{2} \dashrightarrow (\sqrt{2}-1)/2)$. Если a имеет своим сомножителем степень с дробным показателем, то проверяется наличие такой же степени среди "явных" сомножителей знаменателя. Этим предотвращаются нежелательные преобразования перенесения иррациональности в знаменатель, например, $(2-\sqrt{2})/2 \dashrightarrow (\sqrt{2}-1)/\sqrt{2})$. Уровень срабатывания равен 3.

12. Сокращение на общий множитель оснований двух степеней с одинаковыми показателями.

    Обычно общая стандартизация преобразует степень произведения в произведение степеней, а степень дроби - в отношений степеней. Однако, иногда такое преобразование невыполнимо, так как приводит к нарушению о.д.з. (например, к извлечению квадратных корней из отрицательных чисел). Тогда степени произведений и дробей сохраняются, и для работы с ними нужны дополнительные приемы. Один из таких приемов - сокращение двух степеней произведений, имеющих одинаковые показатели:

$$\forall_{abcdefg}\left(\neg(b=0)\rightarrow\frac{a(bc/g)^d}{e(bf)^d}=\frac{a(c/fg)^d}{e}\right)$$

    Указатель "единица" допускает обращение в единицу переменных a, c, e, f, g. Уровень срабатывания равен 1.

13. Сокращение для степени произведения двух чисел и произведения одинаковых степеней этих чисел. Как и предыдущий, данный прием выполняет сокращение для степени произведения:

    $\forall_{abcdpq}(c - \text{rational}\ \&\ \neg(\text{знаменатель}(c) - \text{even})\ \&\ 0 < ab \rightarrow pa^cb^c/q(ab)^d = p(ab)^{c-d}/q)$

    Указатель "дробь" разрешает применять прием с "перевернутыми" числителем и знаменателем. Если выражения a, b не содержат неизвестных, а d содержит, то преобразование блокируется. Уровень срабатывания равен 2.

14. Устранение отрицательного показателя степени в сомножителе числителя либо знаменателя.

$$\forall_{abcd}\left(\frac{a^{-b}c}{d}=\frac{c}{a^bd}\right)$$

$$\forall_{abcd}\left(\frac{a}{b^{-c}d}=\frac{ab^{c}}{d}\right)$$

Уровень срабатывания равен 1.

15. Сложение дробей с одинаковыми знаменателями.

$$\forall_{abc}\left(\frac{a}{b}+\frac{c}{b}=\frac{a+c}{b}\right)$$

Прием применяется в задачах на преобразование. Действие блокируется, если сумма дробей является показателем степени либо аргументом тригонометрической операции: в этих случаях общая стандартизация направлена на получение явных сумм. Оно блокируется также при формальном интегрировании. Уровень срабатывания равен 1. Создана еще одна версия приема по данной теореме - для константных числителей. Как и первая, она блокируется для тригонометрических операций и показателей степени.

В особых случаях, когда сложение дробей с одинаковыми знаменателями дает существенные упрощения, оно выполняется без ограничений:

$$\forall_{abcd}\left(\frac{a+b}{c}+\frac{d-b}{c}=\frac{a}{c}+\frac{d}{c}\right)$$

$$\forall_{abc}\left(\frac{a}{b}-\frac{a+c}{b}=-\frac{c}{b}\right)$$

$$\forall_{abcde}\left(\frac{a(c-d)}{b}+\frac{a(d+e)}{b}=\frac{a(c+e)}{b}\right)$$

Первые два приема имеют уровень срабатывания 0, последний - 1.

16. Раскрывание скобок для суммы с дробью, приводящее к сокращениям, устраняющим дробь:

$$\forall_{abcde}\left(e=\frac{a}{c}\rightarrow a\left(\frac{b}{c}+d\right)=be+ad\right)$$

Антецедент обращается к нормализатору общей стандартизации "нормдробь". Проверяется, что результирующее выражение e не является дробным. Попытка применения приема блокируется, если b/c - константа либо a имеет своим множителем степень суммы. Уровень срабатывания равен 3.

17. Частичное сокращение дроби для упрощения суммы. Если слагаемое числителя дроби делится на знаменатель, то внешнюю сумму иногда удается упростить, вынеся данное слагаемое из числителя наружу и сократив на знаменатель:

$$\forall_{abc}\left(a+\frac{b-ac}{c}=\frac{b}{c}\right)$$

$$\forall_{abc}\left(-a+\frac{ab+c}{b}=\frac{c}{b}\right)$$

$$\forall_{abc}\left(a-\frac{ab+c}{b}=-\frac{c}{b}\right)$$

Первые два приема срабатывают на уровне 2, последний - на уровне 1. Еще один прием этой серии усматривает возможность сложить две числовых константы:

$$\forall_{abcd}\left(\frac{a+bc}{c}+d=\frac{a}{c}+b+d\right)$$

Здесь b, d - десятичные константы. Уровень срабатывания равен 1.

18. Занесение внешнего числового множителя в дробные слагаемые.

    При редактировании ответа задачи на описание, имеющей цель "известно", предпринимается внесение числовых множителей, расположенных перед суммой с дробными слагаемыми, внутрь этой суммы:

    $$\forall_{abcd}\left(d(\frac{a}{b}+c)=\frac{ad}{b}+cd\right)$$

    Переменная d идентифицируется с числовой константой. Уровень срабатывания равен 0.

19. Частичное сокращение дроби при редактировании ответа.

    $$\forall_{abcd}\left(\frac{a+bc}{bd}=\frac{a}{bd}+\frac{c}{d}\right)$$

    Прием применяется при решении функциональных уравнений. Выражение b содержит варьируемую переменную, а выражения c, d не содержат. Таким образом, в найденном ответе выделяется константный член c/d и происходит упрощение варьируемой части. Уровень срабатывания равен 2.

20. Усмотрение подобного слагаемого в числителе дроби. Прием ориентирован на те особые случаи, в которых заблокировано исключение дробных слагаемых числителя. Он усматривает две подобные дроби - среди слагаемых числителя и среди слагаемых внешней суммы:

    $$\forall_{abmnkpq}\left(\frac{a+bm/n}{k}+\frac{bp}{q}=\frac{a}{k}+\frac{(mk+pnk)b}{nkq}\right)$$

    Переменные m, n, k, p, q идентифицируются с константами. Выражение b неконстантное. Уровень срабатывания равен 4.

21. Сокращение дроби с помощью равенства в посылках.

    $$\forall_{abcdn}\left(a-b=0\rightarrow\frac{a^nc}{b^nd}=\frac{c}{d}\right)$$

    Указатель "дробь" обеспечивает применение приема при перестановке числителя и знаменателя. Уровень срабатывания равен 1.

Равенство дроби нулю

$$\forall_{ab}\left(\neg(b=0)\to a=0\leftrightarrow\frac{a}{b}=0\right)$$

Замена выполняется справа налево. Антецедент обрабатывается проверочным оператором. Уровень срабатывания равен 0.

Предусмотрен аналогичный прием для случая, когда равна нулю сумма двух дробей с одинаковыми знаменателями, а сложение этих дробей по каким-то причинам было заблокировано:

$$\forall_{abc}\left(\neg(b=0)\to\frac{a}{b}+\frac{c}{b}=0\leftrightarrow a+c=0\right)$$

Уровень срабатывания равен 2.

Если из контекста не усматривается, что знаменатель имеет ненулевое значение, то усмотрение ненулевого значения дроби обеспечивается следующим приемом:

$$\forall_{ab}\left(\neg(a=0)\to\neg(\frac{a}{b}=0)\right)$$

Фактически прием срабатывает лишь в очень специальной ситуации: при редактировании ответа, когда сопровождающие по о.д.з. посылки уже отброшены, происходит попытка объединить частный случай с общим, а подстановка частного значения обращает в ноль некоторый знаменатель. Уровень срабатывания равен 2.

Уравнение с неизвестной дробью в одной части

Уравнения с неизвестными дробными выражениями обычно преобразуются решателем по следующей схеме: сначала все неизвестные слагаемые группируются в левой части, а известные - в правой, затем выполняется сложение дробей в левой части, и наконец, происходит исключение знаменателя. Последний этап реализуется следующим приемом:

$$\forall_{abcdepqrs}\left(\frac{a}{b}=\frac{dp}{es}\ \&\ c=\frac{pq}{rs}\ \&\ \neg(b=0)\to\frac{a}{b}=c\leftrightarrow p=0\ \vee\ dr=eq\right)$$

Левая часть a/b содержит неизвестные, правая часть c - не содержит. Текущая задача имеет тип "описать" либо "исследовать". Уравнение не обязательно должно быть корневым - оно может располагаться под отрицанием, дизъюнкцией, внутри кванторов или описателей. Однако, если c - переменная, не встречающаяся в левой части и связанная внешним квантором либо описателем, то прием блокируется. Первый и второй антецеденты выделены указателями "идентификатор". Они сначала обращаются к нормализатору "видумножение" для разложения на множители выражений a, b, c. Затем идентифицируются общий множитель p числителей обеих частей уравнения, а также общий множитель s знаменателей. Третий антецедент, выделенный указателем "блокпроверок", проверяет отличие от нуля знаменателя левой части уравнения. После этого оказывается возможным сразу сократить уравнение на s. Заменяющее утверждение представляет собой дизъюнкцию равенства нулю общего множителя p числителей и результата исключения знаменателей частей уравнения после сокращения их на p. При проверке условия $\neg(b=0)$ целесообразно использовать исходное значение b, еще до попытки разложения его на множители. С этой целью введен указатель "копия(фикс(3 1 1))". Обращения к нормализатору "видумножение" несколько ограничены: перед разложением на множители числителя

a проверяется, что он имеет общую переменную с правой частью c; перед разложением на множители знаменателя b - проверяется, что он имеет общую переменную с c и что в c встречаются дроби. Уровень срабатывания равен 1.

На той же самой теореме основан еще один прием, используемый уже в задачах на доказательство. При решении таких задач часть их переменных может искусственным образом объявляться "неизвестными" (они перечисляются в комментарии (неизвестные ...)), чтобы обозначить тенденцию к выражению одних переменных через другие. Тогда и применяется указанный прием. Преобразуемое им равенство расположено в посылках задачи. Уровень срабатывания тоже равен 1.

В задачах на исследование, имеющих цель "известно", не всегда реализуется перегруппировка неизвестных членов уравнения в левую часть. При этом исключение знаменателей тоже выполняется далеко не всегда: велика вероятность того, что соотношение вообще не понадобится. Однако, если получено соотношение без невырожденных числовых атомов (выражений типа "расстояние(AB)", "угол(ABC)", и т.п.), то исключение знаменателей все же выполняется:

$$\forall_{abc}\left(\neg(b=0)\rightarrow\frac{a}{b}=c\leftrightarrow a-bc=0\right)$$

Для срабатывания приема необходимо, чтобы существовала неизвестная, встречающаяся как в a, так и в c. Уровень срабатывания равен 6.

Наконец, упомянем еще один прием для исключения дробей в уравнениях. Он представляет собой общую стандартизацию утверждений и не требует наличия неизвестных:

$$\forall_{abc}\left(\neg(a=0)\rightarrow\frac{a}{b}-\frac{a}{c}=0\leftrightarrow b=c\right)$$

Уровень срабатывания равен 0.

Попытка сложения дробей при упрощении выражений

Если условие задачи на преобразование имеет сумму с дробным слагаемым, то предпринимается попытка упростить ее, преобразовав к виду дроби:

$$\forall_{abcd}\left(d=\frac{a}{b}+c\rightarrow\frac{a}{b}+c=d\right)$$

Антецедент обращается к нормализатору "видумножение". Если преобразуемая сумма представляет собой показатель степени, то сразу после этого к ней применяется нормализатор "стандплюс". Прием срабатывает в задачах на преобразование, имеющих цель "упростить" и не используемых для преобразования подынтегрального выражения. Результат d сложения дробей сравнивается с исходной суммой. Для выполнения преобразования необходимо выполнение одного из следующих условий:

1. Выражение d короче исходной суммы.
2. d - константа, не имеющая вида суммы.
3. d не имеет вида суммы, причем преобразуемое вхождение является сомножителем, операндом дроби либо основанием степени.
4. d не является дробью и не имеет дробного слагаемого.

5. Задача решается для редактирования ответа задачи на описание, имеющей цель "известно", d не имеет обобщенного сомножителя (т.е. операнда, достижимого по цепочке переходов через произведения, дроби и основания степени) вида суммы, а внешней операцией является модуль.

6. Кроме того, в специальных случаях никаких дополнительных условий на d не накладывается. Это происходит, если задача имеет цель "сложитьдроби" либо цель "редакция" (упрощение известных подтермов ответа задачи на описание), либо числитель a содержит конечную сумму, а знаменатель b не содержит.

Введен ограничитель длины выражения d: если преобразуется корневая сумма, то это выражение должно иметь не более 210 символов, иначе - не более 85. Наконец, перед обращением к сложению дробей проверяется выполнение следующих условий:

1. Рассматриваемая сумма не является составной числовой дробью.

2. При наличии комментария "сокращение" учитывается ограничитель сложности: условие задачи на преобразование не должно иметь более 100 символов, а рассматриваемая сумма - более 30. Такой комментарий вводится для ускорения вычислений, если априори ожидается появление достаточно громоздких выражений, у которых упрощение при сложении дробей маловероятно (например, при дифференцировании).

3. При наличии цели "известны", указывающей, что выполняется упрощение ответа задачи на описание, имеющей цель "известно", условие задачи на преобразование должно иметь менее 120 символов.

4. Сумма не является операндом тригонометрической операции.

5. Сумма не расположена внутри другой суммы с дробным слагаемым, к которой еще не применялась попытка сложения дробей.

6. Задача не имеет целей "частнпроизв", "простейшиедроби", "линчастнрешение", "формулатейлора", "раскрытьскобки".

7. Сумма не расположена внутри той части описателя "отображение", которая задает значение функции.

8. Если конечная сумма является сомножителем слагаемого выражения c, то a тоже должно содержать конечную сумму.

9. Сумма не является операндом равенства, расположенного внутри описателя "класс".

Прием срабатывает на уровнях 2, 5 и 7. Ограничения на уровень срабатывания определяются следующими пунктами:

1. Если уровень срабатывания равен 2, то условие задачи не содержит символа "мощность", преобразуемая сумма не имеет вхождений конечных произведений, и ее квадрат не является слагаемым условия.

2. Если уровень срабатывания равен 5, то преобразуемая сумма имеет вхождение конечного произведения.

3. Если уровень срабатывания равен 7, то сумма расположена внутри другой суммы с дробным слагаемым, для которой уже выполнялась попытка сложения дробей.

Для сохранения информации о попытках сложения дробей используется комментарий (короче ...).

Кроме приема сложения дробных выражений в задачах на преобразование, созданы аналогичные приемы, срабатывающие в задачах других типов. Для задач на исследование, имеющих цель "известно", введен прием, реализующий попытку упрощения подсуммы, образованной известными слагаемыми (включая дробное):

$$\forall_{abcdf}\left(d=\frac{a}{b}+c\rightarrow\frac{a}{b}+c+f=d+f\right)$$

Здесь дробь a/b не содержит неизвестных, c идентифицируется как непустая сумма всех остальных известных слагаемых. Антецедент обращается к нормализатору "видумножение", после чего проверяется, что результат d короче исходного выражения. Преобразуемая сумма не расположена под тригонометрической операцией. Имеются две версии приема. Первая срабатывает на уровне 0 и относится к случаю, когда текущая посылка содержит неизвестные. Дополнительно проверяется, что каждый параметр выражения c встречается в a/b и что задача не является "почти решенной" - все неизвестные выражаются из уравнений некоторого простейшего вида. Другая версия срабатывает на уровне 2 и относится к случаю, когда текущая посылка не содержит неизвестных.

Для задач на доказательство и на исследование предусмотрен прием, складывающий неизвестные дробные выражения, если преобразуемая сумма является обобщенным сомножителем числителя либо знаменателя внешней дроби. Уровень срабатывания равен 2.

Еще несколько приемов сложения дробных выражений ориентированы на специальные, редко встречающиеся ситуации.

Разложение на множители для числителя

Разложение на множители числителей и знаменателей дробных выражений, встречающихся в условиях задачи на преобразование, подготавливает возможность сокращения этих выражений и упрощает их сложение. Оно выполняется в подавляющем большинстве случаев, причем по завершении упрощений может быть включен обратный режим раскрывания скобок, если это дает более короткие выражения. Начнем с приемов, преобразующих числитель:

$$\forall_{abcdef}\left(f=b+c\ \&\ d-\mathrm{rational}\ \&\ \neg(\text{знаменатель}(d)-\mathrm{even})\rightarrow\frac{a(b+c)^d}{e}=\frac{af^d}{e}\right)$$

$$\forall_{abcdef}\left(f=b+c\ \&\ 0\leq b+c\rightarrow\frac{a(b+c)^d}{e}=\frac{af^d}{e}\right)$$

Фактически разложению на множители подлежат суммы - основания степени сомножителей числителя. Разделение приема на два подслучая - для рациональных показателей степени, имеющих нечетный знаменатель, и для прочих показателей - вызвано необходимостью обеспечить во втором подслучае сопровождение по о.д.з.

Здесь указатель "вывод(меньшеилиравно(0 x6)эквивалентно(2))" сопровождает замену выводом посылки $0 \leq f$, необходимой для дальнейшей работы со степенью f^d. Приемы применяются для преобразования дроби, входящей в условие задачи на преобразование, имеющей цель "упростить". Антецедент обращается к нормализатору "видумножение". Для срабатывания первой версии приема необходимо выполнение следующих дополнительных условий:

1. Отсутствует комментарий (Слагаемое $b{+}c$), блокирующий попытку разложения на множители после того, как выполнялось раскрывание скобок.
2. Дробь не расположена под описателем "отображение".
3. Если преобразуется подынтегральное выражение и сумма $b{+}c$ не имеет дробных слагаемых, то произошло сокращение дроби на неконстантный множитель.
4. Если редактируется ответ задачи на описание, имеющей цель "известно", то сумма $b + c$ имеет не более 120 символов, причем b, c имеют общий параметр.
5. Дробь не является слагаемым показателя степени.
6. Выражение $b + c$ не содержит конечных сумм.
7. Если f представляет собой сумму, то она не должна содержать степеней с дробным показателем, отсутствовавших в условии задачи до преобразования.
8. Если f имеет более 160 символов, то сумма $b + c$ имела не менее 120 символов.

Для срабатывания второй версии необходимо выполнение пунктов 1,2,6. При этом проверяется, что d не является простой дробью с нечетным знаменателем. Если редактируется ответ задачи на описание, имеющей цель "известно", то проверяется, что результат разложения на множители короче исходного выражения. Уровни срабатывания обеих версий равны 3.

Каждый из рассмотренных двух приемов продублирован для уровня срабатывания 5, где условия применения несколько ослаблены. Кроме того, созданы несколько приемов, обращающихся к разложению на множители числителя в задачах, не имеющих типа "преобразовать". Они ориентированы на особые, редко встречающиеся ситуации.

Разложение на множители для знаменателя

Приемы аналогичны рассмотренным в предыдущем разделе:

$\forall_{abcdef}(\neg(b + c = 0) \;\&\; f = b + c \;\&\; d - \mathrm{rational} \;\&\; \neg(\text{знаменатель}(d) - \mathrm{even}) \rightarrow a/((b + c)^d e) = a/(f^d e))$

$\forall_{abcdef}(0 < b + c \;\&\; f = b + c \rightarrow a/((b + c)^d e) = a/(f^d e))$

Для каждой теоремы созданы две версии приема; одна срабатывает на уровне 3, другая (с ослабленными фильтрами) - на уровне 5.

Устранение квадратичной иррациональности в знаменателе

Для устранения суммы с иррациональностью - основания степени сомножителя знаменателя - используется домножение числителя и знаменателя на сопряженную сумму:

$\forall_{abcdefgpq}(\neg(a+b=0)\ \&\ g=a-b\ \&\ p=a^2\ \&\ q=b^2\ \&\ c=p-q\ \&\ (\neg(g=0)\ \vee\ \neg(c=0))\rightarrow f/(d(a+b)^e)=f(g/c)^e/d)$

Прием применяется в условии задачи на преобразование. Устраняемой суммой является $a+b$. Фильтры приема уточняют ее вид: сумма должна быть неконстантной (для константных сумм созданы свои приемы, приведенные ниже), причем a должно иметь сомножителем степень с дробным показателем $m/2n$. В основании R этой степени должна встречаться сумма, достижимая из корня R только через алгебраические операции $+,-,\times,/$, а также через основания степеней. Если выражение b имеет слагаемое, делящееся на некоторую степень с дробным показателем и неконстантным основанием, то оно должно состоять только из этого слагаемого. Первый антецедент проверяет отличие исключаемой суммы от 0. Он нужен, чтобы в комментариях сохранилась информация о выводимости утверждений. Второй антецедент формирует сопряженную сумму g. Третий и четвертый антецеденты возводят в квадрат выражения a,b, используя при этом нормализатор раскрывания скобок "стандплюс". Пятый антецедент упрощает разность квадратов c с помощью нормализаторов "нормплюс", "видумножение". Чтобы преобразование было корректным, необходимо убедиться в отличии от нуля значения сопряженной суммы g. Иногда это легко сделать сразу, но в некоторых случаях такая проверка может столкнуться с трудностями, в то время как отличие от нуля разности квадратов c, тоже гаранитрующее корректность, будет усматриваться без труда. Поэтому шестой антецедент имеет вид дизъюнкции условий $\neg(g=0),\neg(c=0)$. Он выделен указателем "блокпроверок", а компилятор организует обращения к проверочному оператору "усмне0" для каждого дизъюнктивного члена по отдельности. После упрощения разности квадратов c проверяется, что она не имеет слагаемых, делящихся на степени с показателями вида $m/2n$. Указатель "вывод(не(равно(х3 0))эквивалентно(1))" обеспечивает необходимый для сопровождения по о.д.з. вывод утверждения $\neg(c=0)$. Созданы две версии приема. Одна срабатывает при отсутствии дробей, достижимых из корня выражения $a+b$ через операции $+,-,\times,/$ и через основания степеней, другая - при наличии таких дробей. В первом случае уровень срабатывания равен 4, во втором - 5. Первая версия, в свою очередь, подразбита на два приема, имеющих одинаковые уровни срабатывания. Первый из них применяется в отсутствии цели "нормИнтнграл", второй - при ее наличии. Эта цель указывает на режим преобразования подынтегрального выражения при вычислении неопределенного интеграла. Проверяется, что переменная интегрирования x встречается в сумме $a+b$ и либо не входит в c, либо f,c могут быть сокращены на общий множитель, содержащий x. Проверяется также, что рассматриваемое дробное выражение не является слагаемым суммы, имеющей другое дробное выражение вида $A/(B(a+b)^k)$.

Если проверка отличия от нуля выражений g,c затруднена, то применяется еще одна версия приема:

$\forall_{abcdefghpqmn}(\neg(a+b=0)\ \&\ g=a-b\ \&\ p=a^2\ \&\ q=b^2\ \&\ c=p-q\ \&\ 0<e\ \&\ f=mh\ \&\ hn=c^e\ \&\ \neg(n=0)\rightarrow f/(d(a+b)^e)=mg^e/dn)$

Показатель степени e идентифицируется с простой дробью, имеющей нечетный знаменатель, причем усматривается возможность сокращения c^e и f на общий множитель h. Тогда достаточно усмотреть отличие от нуля выражения n. Фильтры - те же, что и выше; уровень срабатывания равен 4. Наконец, еще две версии созданы для использования их на этапе редактирования ответа, когда сопровождение по о.д.з. может оказаться частично нарушенным и усмотреть отличие от нуля суммы $a+b$ не удастся. Они отличаются от рассмотренных только отбрасыванием первого антецедента.

В случае задач на описание создан прием, устраняющий иррациональность в знаменателе, если при этом происходит упрощение уравнения относительно неизвестных:

$\forall_{abcpqrms}(m = a^2b - c^2 \ \& \ \neg(m = 0) \ \& \ s = p/qm^n \rightarrow p(a\sqrt{b} - c)^n/(q(a\sqrt{b} + c)^n) = r \leftrightarrow s(a\sqrt{b} - c)^{2n} = r)$

Выражение $a\sqrt{b} - c$ содержит неизвестные; правая часть r и результат s деления остальных множителей числителя на новый знаменатель неизвестных не содержат. Уровень срабатывания равен 0.

Все перечисленные выше приемы относились к выражениям общего вида, причем во многих из них явно оговаривалось, что сумма с радикалами должна быть неконстантной. Для константных сумм предусмотрены два отдельных приема:

$$\forall_{abcdefghp}\left(p = c^2e^g - d^2f^h \ \& \ \neg(p = 0) \rightarrow \frac{a}{b(ce^{g/2} + df^{h/2})} = \frac{a(ce^{g/2} - df^{h/2})}{bp}\right)$$

$$\forall_{abcdefg}\left(g = c^2 - d^2e^f \ \& \ \neg(g = 0) \rightarrow \frac{a}{b(c + de^{f/2})} = \frac{a(c - de^{f/2})}{bg}\right)$$

В первом приеме c, d, e, f идентифицируются с десятичными константами; g, h - с натуральными константами, меньшими 5. Первый антецедент, выделенный указателем "программа", реализует арифметические вычисления. Во втором приеме c, d, e идентифицируются с десятичными константами, f - с натуральной константой, меньшей 5. Уровень срабатывания в обоих случаях равен 2.

Устранение кубической иррациональности в знаменателе

Если в знаменателе встречается трехчлен с кубическими радикалами, то реализуется попытка упрощения путем домножения числителя и знаменателя до суммы либо разности кубов:

$\forall_{abcdefgpq}(\neg(a^2 - ab + b^2 = 0) \ \& \ g = a + b \ \& \ p = a^3 \ \& \ q = b^3 \ \& \ c = p + q \ \& \ (\neg(g = 0) \ \vee \ \neg(c = 0)) \rightarrow f/(d(a^2 - ab + b^2)^e) = f(g/c)^e/d)$

$\forall_{abcdefgpq}(\neg(a^2 + ab + b^2 = 0) \ \& \ g = a - b \ \& \ p = a^3 \ \& \ q = b^3 \ \& \ c = p + q \ \& \ (\neg(g = 0) \ \vee \ \neg(c = 0)) \rightarrow f/(d(a^2 + ab + b^2)^e) = f(g/c)^e/d)$

Приемы применяются к условию задачи на преобразование, имеющей цель "упростить". Их фильтры аналогичны случаю квадратичной иррациональности. Уровни срабатывания равны 6.

Устранение числовых знаменателей слагаемых уравнения

Если уравнение имеет неизвестное дробное слагаемое, знаменателем которого служит десятичная константа, то выполняется домножение его на эту константу:

$\forall_{abcd}(\neg(c=0)\rightarrow a+b/c=d\leftrightarrow ac+b=cd)$

Текущая задача имеет тип "описать" либо "исследовать"; число ее неизвестных должно быть равно 1. b содержит неизвестные, c - десятичная константа. Каждое дробное слагаемое выражения a должно иметь своим знаменателем десятичную константу. Уровень срабатывания равен 0.

Сложение неизвестных дробей в одной из частей уравнения

$$\forall_{abcde}\left(e=\frac{a}{b}+c\rightarrow\frac{a}{b}+c=d\leftrightarrow e=d\right)$$

Прием применяется к уравнению - условию задачи на описание. Дробь a/b содержит неизвестные; противоположная часть уравнения d известна. Антецедент обращается для сложения дробных выражений к нормализатору "видумножение". Если задача имеет единственную неизвестную, причем уравнение не содержит неизвестных логарифмов от мультипликативных выражений ("умножение", "дробь", "степень") либо логарифмов, имеющих различные основания, то уровень срабатывания равен 1, иначе он равен 3. Для срабатывания приема необходимо выполнение следующих требований:

1. Уравнение не должно содержать неизвестных логарифмов, под которыми расположена сумма с дробными слагаемыми. В этой ситуации сначала будут складываться дроби под логарифмами.
2. Отсутствует другое условие задачи, в котором некоторое равенство, достижимое из корня только через символы "и", "существует", выражает одну из встречающихся в преобразуемом уравнении неизвестных x через другие неизвестные.

При преобразовании дифференциального уравнения накладывается ряд дополнительных ограничений на применение данного приема, причем специально для этого случая добавлена еще одна его версия, срабатывающая на уровне 6.

Для задач на исследование, имеющих цель "известно", созданы несколько приемов сложения дробей в левой части уравнения, срабатывающих на уровне 6. В преобразуемую сумму не должны входить невырожденные числовые атомы. Либо знаменатель дроби не содержит неизвестных, либо правая часть уравнения нулевая. К геометрическим задачам эти приемы не применяются.

Сложение известных дробей при проверке сопровождающего тождества

На завершающем этапе решения уравнения или системы уравнений могут появиться равенства без неизвестных, для которых выполняется проверка - упрощение их до получения логической константы либо ограничений на параметры задачи. Одним из упрощающих приемов здесь является сложение дробных выражений в уравнении с нулевой правой частью:

$$\forall_{abcd}\left(d=\frac{a}{b}+c\rightarrow\frac{a}{b}+c=0\leftrightarrow d=0\right)$$

Задача на описание должна иметь цель "учетответа". Преобразуется равенство без неизвестных, построенное из переменных и числовых констант с помощью простейших операций $-,+,/,\times$ и возведения в степень. Уровень срабатывания равен 1.

Домножение уравнения на известный знаменатель слагаемого левой части

$$\forall_{abcd}\left(\neg(c=0)\ \&\ d-\text{число}\rightarrow a+\frac{b}{c}=d\leftrightarrow ac+b=cd\right)$$

Созданы две версии приема. Одна из них применяется к задачам на доказательство либо на исследование, другая - к задачам на описание, имеющим цель "независит X". c не содержит неизвестных. В первом случае проверяется, что равенство не выражает один числовой атом через другие, во втором - что запрещенные переменные X не входят в c и что равенство не выражает одну неизвестную через другие. Уровень срабатывания равен 2.

Дополнительно созданы два приема, ослабляющие ограничения на срабатывание, если левая часть уравнения сама является дробью:

$$\forall_{abc}\left(\neg(b=0)\rightarrow\frac{a}{b}=c\leftrightarrow a=bc\right)$$

В обоих приемах a,c содержат неизвестные, b - не содержит. Первый прием требует, чтобы a содержало невырожденный негеометрический числовой атом. Если при этом правая часть недробная, то каждый ее невырожденный числовой атом входит в левую часть. Если правая часть - неизвестная, то a не делится на другую неизвестную. Уровень срабатывания первого приема равен 2. Второй прием требует, чтобы уравнение не содержало невырожденных числовых атомов. При этом задача - негеометрическая; если c - переменная, то она входит в левую часть. Уровень срабатывания равен 7.

Уравнения в задачах на исследование, имеющих цель "известно"

Для преобразования дробных выражений в задачах на исследование, имеющих цель "известно", созданы следующие приемы:

1. Сокращение числителей на общий множитель.

   $$\forall_{abcd}\left(a=pe\ \&\ c=pf\ \&\ \neg(p=0)\rightarrow\frac{a}{b}=\frac{c}{d}\leftrightarrow\frac{e}{b}=\frac{f}{d}\right)$$

   Первый и второй антецеденты обрабатывают числители a,c нормализатором "факторизация" и определяют общий множитель p. Перед этим проверяется, что a имеет своим сомножителем сумму. Прием применяется не только в задачах на исследование, но и в посылках задач на доказательство. Уравнение не должно содержать неизвестных тригонометрических операций. Уровень срабатывания равен 3.

2. Сокращение знаменателей на общий множитель.

   $$\forall_{abcd}\left(b=pe\ \&\ d=pf\rightarrow\frac{a}{b}=\frac{c}{d}\leftrightarrow\frac{a}{e}=\frac{c}{f}\right)$$

   Антецеденты определяют общий множитель p, не обращаясь к нормализаторам. Проверяется, что этот множитель содержит неизвестную, не входящую в результирующее равенство. Уровень срабатывания равен 3.

3. Упрощение многоэтажных дробей. Если задача имеет посылку, выражающую неизвестную x в виде дроби, причем знаменатель последней содержит сумму с дробным выражением, то дробь упрощается при помощи вспомогательной задачи на преобразование:

$$\forall_{abcx}\left(c = \frac{a}{b} \to x = \frac{a}{b} \leftrightarrow x = c\right)$$

   Выражения a, b содержат неизвестные; уравнение не имеет невырожденных числовых атомов. Проверяется, что c либо не содержит неизвестных, либо короче исходной дроби. Уровень срабатывания равен 3.

4. Сокращение дроби.

   Соотношения, выписываемые в процессе решения задачи на исследование, имеющей цель "известно", большей частью бывают нужны лишь для того, чтобы зафиксировать наличие связей между числовыми атомами. Лишь малая их доля фактически используется в цепочке вычислений, приводящей к ответу. Поэтому все сколь-нибудь трудоемкие алгебраические преобразования в посылках таких задач заблокированы. Они будут применяться к выделяемым специальными приемами подсистемам уравнений, решаемым вспомогательными задачами. В частности, сокращение дробей, связанное с попытками разложения на множители, применяется лишь в самых простых случаях. На уровне 1 срабатывают следующие приемы:

$$\forall_{abcd}\left(\frac{ab}{ac + ad} = \frac{b}{c + d}\right)$$

$$\forall_{abcd}\left(\frac{ab + ae}{ac + ad} = \frac{b + e}{c + d}\right)$$

   На уровне 5 попытка сокращения дроби предпринимается, если числитель и знаменатель имеют своими сомножителями степени сумм:

$$\forall_{abcd}\left(a = \frac{b}{c} \to \frac{b}{c} = d \leftrightarrow d = a\right)$$

   Антецедент обрабатывает числитель и знаменатель нормализатором "видумножение", после чего выполняет сокращение с помощью нормализатора "нормдробь". Замена выполняется, если b, c содержат неизвестные, а полученное после сокращения выражение a неизвестных не содержит. Правая часть d должна содержать неизвестные. Прием имеет достаточно сильный ограничитель трудоемкости.

   Наконец, на уровне 8 предпринимается попытка усмотреть делимость числителя дроби на константный знаменатель:

$$\forall_{abc}\left(a = bc \;\&\; \neg(b = 0) \to \frac{a}{b} = c\right)$$

   Числитель a должен представлять собой неконстантную сумму; b - константное выражение. Антецедент обрабатывает a нормализатором "факторизация" и усматривает его делимость на b.

5. Определение значения неизвестной дроби из соотношения пропорциональности. Если в посылках явно указано соотношение пропорциональности для двух неизвестных выражений, то оно позволяет найти значение частного данных выражений:

$$\forall_{abcdmn}\left(\neg(a=0)\ \&\ \neg(y=0)\ \&\ \frac{ax}{b}=\frac{cy}{d}\rightarrow\frac{mx}{ny}=\frac{mbc}{nad}\right)$$

$$\forall_{abxymn}\left(ax+by=0\ \&\ \neg(a=0)\rightarrow\frac{mx}{ny}=-\frac{bm}{an}\right)$$

Оба приема выполняют преобразование замены. Первый из них срабатывает на уровне 3. Его третий антецедент идентифицируется с посылкой. Общие множители x, y операндов дроби и частей соотношения пропорциональности содержат неизвестные; выражения a, b, c, d - не содержат. Текущая посылка задачи не имеет невырожденных числовых атомов. Второй прием срабатывает на уровне 5. x, y идентифицируются с непустыми произведениями всех неизвестных множителей числителя и знаменателя.

6. Блокировка попытки явного разрешения относительно неизвестной вырожденного уравнения. Прием блокирует попытку разрешения уравнения относительно единственной неизвестной, если оно после сокращения обеих частей на общий неизвестный множитель x превращается в соотношение для известных параметров:

$$\forall_{abcdx}\left(\frac{ax}{b}=\frac{cx}{d}\rightarrow\emptyset\right)$$

Прием имеет заголовок "замечание". Блокировка обеспечивается вводом комментария "неизвестная" к рассматриваемому уравнению. a, b, c, d известны. Уровень срабатывания равен 1.

7. Использование соотношения пропорциональности для уменьшения числа неизвестных в дроби.

$$\forall_{abcdmnpq}\left(\neg(p=0)\ \&\ \frac{m}{an}=\frac{p}{bq}\rightarrow x=\frac{ac}{bd}\leftrightarrow x=\frac{mqc}{npd}\right)$$

x идентифицируется с неизвестной внешней задачи на описание; правая часть преобразуемого равенства не содержит x. Второй антецедент - соотношение пропорциональности, непосредственно находящееся в посылках. Оно тоже не содержит переменной x. Общие множители a, b операндов дроби и знаменателей соотношения пропорциональности содержат неизвестные. Число неизвестных в правой части результирующего равенства меньше числа неизвестных в правой части исходного равенства. Уровень срабатывания равен 5.

$$\forall_{abcdpqn}\left(\neg(c=0)\ \&\ ac=bd\rightarrow\frac{pa^n}{db^n}=\frac{pd^n}{qc^n}\right)$$

Выражения a, b содержат неизвестные; c, d - не содержат. Уровни срабатывания равны 3 и 5.

8. Исключение знаменателей. Это преобразование применяется с сильными ограниченями. Прежде всего, отметим случай равенства знаменателей частей уравнения:

$$\forall_{abc}\left(\neg(a=0)\to\frac{b}{a}=\frac{c}{a}\leftrightarrow b=c\right)$$

$$\forall_{abc}\left(\neg(a=0)\to-\frac{b}{a}=\frac{c}{a}\leftrightarrow -b=c\right)$$

Уровни срабатывания равны 4.

$$\forall_{abc}\left(\neg(b=0)\to\frac{a}{b}=c\leftrightarrow a-bc=0\right)$$

Этот прием применяется, если все слагаемые числителя a и правой части c имеют общий ненулевой неизвестный множитель. При этому уравнение не имеет невырожденных числовых атомов. Уровень срабатывания равен 4. На той же самой теореме основаны еще два приема, срабатывающие на уровне 5. Первый из них применяется, если знаменатель дроби содержит невырожденный числовой атом, а числитель и правая часть не содержат. Второй - если числитель имеет невырожденный числовой атом, знаменатель и правая часть таких атомов не имеют, причем равенство не выражает неизвестную c через известные параметры. Последний прием продублирован; вторая его версия срабатывает на уровне 9. Обе версии в геометрических задачах не применяются.

$$\forall_{abcd}\left(\neg(b=0)\ \&\ \neg(d=0)\to\frac{a}{b}=\frac{c}{d}\leftrightarrow ad-bc=0\right)$$

Этот прием применяется в задачах аналитической геометрии, связанных с системами координат. Должна иметься посылка, содержащая символ "коорд". Уравнение не имеет невырожденных числовых атомов; количество числовых неизвестных задачи не менее 4, и имеется не менее 5 уравнений без невырожденных числовых атомов. Уровень срабатывания равен 8.

9. Частичное сокращение дроби.

$$\forall_{abcd}\left(\frac{ab+c}{a}=b+\frac{c}{a}\right)$$

Дробь является слагаемым одной из частей уравнения. Выражение b содержит неизвестные, причем существует слагаемое той же или противоположной части уравнения, имеющее вид kb, где k известно. Уровень срабатывания равен 4.

10. Ввод вспомогательного параметра - числовой вспомогательной неизвестной, являющейся множителем знаменателя дроби. Если искомая величина представлена в виде дроби a/by, где y - вспомогательная неизвестная, то можно временно рассматривать y как известный параметр. Это инициирует попытки выразить через него прочие неизвестные, входящие в числитель, и в конечном итоге вероятно сокращение дроби на y. Прием имеет следующую теорему:

$$\forall_{abcxy}\left(x=\frac{a}{by}\to c=y\right)$$

x идентифицируется с неизвестной внешней задачи на описание Z, y - с неизвестной задачи на исследование, не являющейся неизвестной Z (она входит в посылки задачи Z, но не должна остаться в ответе). Указатель "вспомпараметр(x3 фикс(0))" обеспечивает выбор новой переменной c в качестве вспомогательного параметра. Она будет зарегистрирована в комментарии (вспомпараметр ...). После того, как появятся выражения для всех неизвестных внешней задачи (с возможным участием c), эта переменная будет переведена в категорию неизвестных, и решение задачи продолжится. Уровень срабатывания равен 13.

11. Нахождение частного квадратов величин, для которых имеется связывающее их модули соотношение пропорциональности.

$$\forall_{abcdpq}\left(\neg(d=0)\ \&\ d|a|=c|b|\to\frac{a^2p}{b^2q}=\frac{c^2p}{d^2q}\right)$$

Второй антецедент идентифицируется с посылкой. c,d известны; a,b содержат неизвестные. Прием выполняет замену, извлекая из посылки выражение для частного квадратов a,b. Уровень срабатывания равен 3.

12. Перенесение степени десяти из знаменателя в числитель при редактировании ответа. Прием выполняет преобразование ответа к стандартному виду домножения на степень десяти, принятому в задачах по физике и химии:

$$\forall_{abn}\left(\frac{a}{b10^n}=\frac{a10^{-n}}{b}\right)$$

Прием срабатывает в задачах на преобразование, имеющих цель "учетрезультата". Такие задачи выполняют редактирование ответа задач на описание, имеющих цель "известно". Переменная n идентифицируется с десятичной константой. Уровень срабатывания равен 2.

13. Переход к приближенной десятичной дроби. Это еще один прием редактирования дробного ответа задачи на описание, имеющей цель "известно". Простые дроби в задачах по химии заменяются приближенными десятичными константами. Вычисление выполняется математическим сопроцессором, так что первоначально выдается достаточно большое число цифр. Округление делается другим приемом.

$$\forall_{abcpq}\left(c=\frac{a}{b}\to\frac{ap}{bq}=\frac{cp}{q}\right)$$

a,b - десятичные константы. Фильтр "Разделы(химия)" проверяет наличие в задаче понятий, относящихся к химии; фильтр "не(Разделы(механика))" блокирует прием, если задача относится к механике. Указатель "выч(1)" определяет обработку первого антецедента с помощью вычислений в формате "с плавающей запятой". Уровень срабатывания равен 2.

Сложение дробей в уравнениях, описывающих геометрическое место точек

Обычно такие уравнения приводятся к стандартному виду без дробных слагаемых:

$$\forall_{abcd}\left(d=\frac{a}{b}+c\rightarrow\frac{a}{b}+c=0\leftrightarrow d=0\right)$$

Антецедент обращается к вспомогательной задаче на упрощение, а также к нормализатору "видумножение". Прием применяется в задачах на преобразование, имеющих цель "класс". Эти задачи преобразуют координатное задание множества точек к бескоординатному виду. Рассматриваемая сумма расположена под квантором существования, связывающим некоторые ее параметры. Уровень срабатывания равен 3.

Коррекция сопровождения по о.д.з.

В некоторых ситуациях сопровождение по о.д.з. в задаче может нарушиться. Хотя все условия на о.д.з. будут следствиями утверждений задачи, но это станет неочевидным, и механизмы быстрой проверки корректности преобразований работать не смогут. Чтобы исправить это положение, может быть введен комментарий "коррекцияодз", инициирующий пополнение сопровождающих утверждений. Например, если в задаче встречается дробь, для которой неочевидно отличие от нуля знаменателя, то соответствующее утверждение заносится в задачу. Теорема этого приема такова:

$\forall_a(\neg(a=0))$

Текущая задача - на описание и имеет комментарий "коррекцияодз". Указатель "контрольвывода(дробь(x2 x1))" определяет активизацию приема при усмотрении дроби b/a в условии задачи. Если проверочный оператор не усматривает отличие a от нуля, то выводится новое условие $\neg(a=0)$. Проверяется, что дробь не расположена внутри дизъюнкции, конъюнкции, условного выражения, квантора либо описателя. Указатель "коррекцияодз" определяет коррекцию комментариев "сопровождение". Уровень срабатывания равен 1.

Преобразование к виду суммы простейших дробей

Преобразование рационального дробного выражения к виду суммы простейших дробей выполняется либо задачами на преобразование, имеющими цель "простейшиедроби", либо задачами на преобразование подынтегрального выражения при формальном интегрировании. Необходимые вычисления реализуются нормализатором "простейшиедроби", описываемым ниже. Пока рассмотрим лишь приемы, обращающиеся к этому нормализатору. Их уровень срабатывания равен 3. Первый прием имеет следующую теорему:

$$\forall_{abc}\left(a=\frac{b}{c}\rightarrow\frac{b}{c}=a\right)$$

Решается задача на преобразование, имеющая цель "простейшиедроби". Преобразуемая дробь - условие задачи. Антецедент обращается к нормализатору "простейшиедроби". Предварительно идентифицируется переменная d, входящая в дробное выражение, которая передается нормализатору через его комментарий (переменная d). Для применения приема необходимо, чтобы результат a имел заголовок "плюс".

При формальном интегрировании обращение к нормализатору "простейшиедроби" происходит в двух случаях - если дробь представляет собой частное двух многочленов либо отличается от него домножением на экспоненту:

$$\forall_{abp}\left(p = \frac{a}{b} \rightarrow \frac{a}{b} = p\right)$$

$$\forall_{abcp}\left(p = \frac{a}{b} \rightarrow \frac{a\mathrm{exp}d}{b} = p\mathrm{exp}d\right)$$

В обоих случаях предполагается выполнение следующих требований:

1. Решается задача на преобразование, имеющая цель "нормИнтеграл"; комментарий (нормИнтеграл x) задает переменную интегрирования.
2. Переход от корня условия задачи к рассматриваемой дроби осуществляется только через операции "плюс", "минус".
3. Знаменатель b имеет своим сомножителем степень (возможно, с показателем единица) суммы, зависящей от x.
4. a, b суть многочлены от x, причем либо a не константа, либо степень b больше единицы.
5. Дробь a/b не является простейшей. Фактически проверяется, что знаменатель нельзя представить в виде $kp(x)^n$, где k - константа, n - натуральное, $p(x)$ - многочлен первой либо второй степени, большей, чем степень числителя.
6. b не имеет сомножителя вида $p(x)^n$, где степень $p(x)$ больше 2.

Нормализатор общей стандартизации "нормдробь"

В нормализаторе представлены следующие приемы упрощения дробей, большинство из которых уже рассмотрены выше как приемы сканирования задачи:

1. Дроби с числителем 0.
2. Дроби со знаменателем 1.
3. Вынесение минуса из знаменателя.
4. Вынесение минуса из числителя.
5. Сокращение числовой дроби.
6. Сокращение численных множителей числителя и знаменателя.
7. Сокращение числовых оснований степеней с одинаковыми показателями, расположенных в числителе и знаменателе.
8. Деление на дробь.
9. Деление дроби.
10. Сокращение дробного выражения (без попыток разложения на множители его числителя и знаменателя).

11. Деление степеней с одинаковыми основаниями.

12. Сокращение на общий множитель оснований двух степеней с одинаковыми показателями.

13. Группировка модулей в числителе и знаменателе.

$$\forall_{abcd}\left(\frac{d|b|}{a|c|}=\frac{d|b/c|}{a}\right)$$

14. Деление на модуль дроби.

$$\forall_{abc}\left(b-\text{число}\ \&\ \neg(c=0)\ \&\ \neg(b=0)\ \&\ c-\text{число}\to\frac{a}{|b/c|}=a|\frac{c}{b}|\right)$$

15. Устранение отрицательного показателя степени в сомножителе числителя либо знаменателя.

16. Степень с дробным знаменателем - сомножитель знаменателя.

$$\forall_{abcd}\left(0<cd\to\frac{a}{b(c/d)^e}=\frac{a(d/c)^e}{b}\right)$$

17. Частичное сокращение для степени с дробным основанием - сомножителя числителя дроби.

    $\forall_{abcdefg}(f-\text{rational}\ \&\ \neg(\text{знаменатель}(f)-\text{even})\ \&\ \neg(c=0)\to ab^f(cd/b)^e/d^fg=ac^f(cd/b)^{e-f}/g)$

    $\forall_{abcdefg}(f-\text{rational}\ \&\ \neg(\text{знаменатель}(f)-\text{even})\ \&\ \neg(d=0)\to ac^f(b/cd)^e/b^fg=a(b/cd)^{e-f}/gd^f)$

    $\forall_{abcde}(\neg(b=0)\to d(a/b)^c/ae=d(a/b)^{c-1}/be)$

    В последнем приеме a имеет заголовок "умножение", b не имеет такого заголовка и короче, чем a.

18. Регистрация во внешнем контексте информации о неудачной попытке сокращения дроби.

    Прием используется при формальном интегрировании. Если не удается сократить дробь вида ac/bc, причем интеграл содержит параметры, общие с выражением c, то целесообразно рассмотрение двух случаев: $c=0$ и $\neg(c=0)$. Прием передает информацию об этом внешнему нормализатору "нормИнтеграл", выполняющему интегрирование. Теорема приема имеет вид:

    $\forall_{abc}(\text{контекст}(ac/bc))$

    Проверяется, что отличие c от нуля не усматривается и что отсутствует посылка $c=0$. Проверяется, что дробь неконстантная. Заголовок приема - "замена(замечание нормдробь)". Выполняемое приемом действие определяется указателем "нормализация(нормИнтеграл условие(пересекаются(параметры(внешкорень)x3))нормили или(равно(x3 0)не(равно(x3 0))))". Внешнему нормализатору "нормИнтеграл" передается комментарий (нормили $c=0\ \vee\neg(c=0)$).

19. Группировка под общий показатель степени при вычислении предела.

$$\forall_{abcdefghi}\left((j \to k\backslash l) \to \frac{ha^{dg+f}}{ie^{bg+c}} = \frac{ha^{f}(a^{d}/e^{b})^{g}}{ie^{c}}\right)$$

В посылках усматривается утверждение о "стремлении" для параметра j. Этот параметр входит в g и не входит в d, f, b, c. Предпринимается группировка двух показательных зависимостей в одну. Прием применяется при наличии комментария "нормпредел", указывающего на наличие внешнего нормализатора вычисления пределов.

20. Устранение квадратичной иррациональности в знаменателе для числовых выражений (только прием для одного радикала).

21. Символы бесконечности.

$$\forall_{a}\left(0 < a \;\&\; a - \text{число} \to \frac{\infty}{a} = \infty\right)$$

$$\forall_{a}\left(0 < a \;\&\; a - \text{число} \to \frac{-\infty}{a} = -\infty\right)$$

$$\forall_{a}\left(a - \text{число} \to \frac{a}{\infty} = 0\right)$$

$$\forall_{a}\left(a - \text{число} \to \frac{a}{-\infty} = 0\right)$$

22. Деление константных выражений с радикалами.

$$\forall_{abcdemnp}\left(p = d^2 - e^2c \;\&\; \neg(p = 0) \to \frac{(a + b\sqrt{c})m}{(d + e\sqrt{c})n} = \frac{(ad - bce + (bd - ae)\sqrt{c})m}{pn}\right)$$

Дробь константная, c - десятичное число. Выражения d, e не содержат степеней. Первый антецедент обрабатывает выражение нормализаторами общей стандартизации и раскрывает скобки.

23. Константные суммы с дробями в числителе либо в знаменателе.

$$\forall_{abcde}\left(\neg(c = 0) \to \frac{(a + b/c)e}{d} = \frac{(ac + b)e}{cd}\right)$$

$$\forall_{abcde}\left(\neg(c = 0) \to \frac{d}{(a + b/c)e} = \frac{cd}{(ac + b)e}\right)$$

Нормализуемые дроби константные.

24. Выражения с "O".

$\forall_{an}((x \to 0\backslash b) \;\&\; \neg(a = 0) \to O(x^n)/a = O(x^n))$

Переменная x не входит в a; n натуральное.

25. Сокращение синуса двойного аргумента.

$$\forall_{abcd}\left(b = 2d \rightarrow \frac{a\sin b}{c\cos d} = \frac{2a\sin d}{c}\right)$$

$$\forall_{abcd}\left(b = 2d \rightarrow \frac{a\sin b}{c\sin d} = \frac{2a\cos d}{c}\right)$$

Приемы применяются только в случае, когда текущая задача - на исследование и имеет цель "известно".

Нормализатор сокращенной записи "упрощдробь"

Для преобразования дробных выражений к компактному виду используются следующие приемы:

1. Группировка под общий показатель степени.

   $\forall_{abcdefg}(c -$ число $\&\ g -$ rational $\&\ \neg($знаменатель$(g) -$ even$)\ \&\ f -$ rational $\&\ \neg($знаменатель$(f)-$even$)\ \&\ \neg($знаменатель$(a)-$even$)\ \&\ a-$rational $\&\ e-$число $\rightarrow db^{af}/ec^{ag} = d(b^f/c^g)^a/e)$

   Прием применяется без ограничений на уровне 3.

   $$\forall_{abcd}\left(b = c^2 \rightarrow \frac{da^2}{b} = d(\frac{a}{c})^2\right)$$

   Здесь b - натуральная константа, для которой усматривается, что она является квадратом натурального числа c. Числитель дроби и выражение a не являются десятичными константами. a не имеет заголовка "степень". Уровень срабатывания равен 1.

   $\forall_{abcdefghij}(0 < a\ \&\ 0 \leq d \rightarrow cd^{ej/fg}/ia^{bj/fh} = c(d^{e/g}/a^{b/h})^{j/f}/i)$

   Допускаются вырожденные единичные значения параметров $b, c, e, f, g, h, , i, j$. При этом f, j одновременно единице не равны. Преобразование очень громоздких константных дробей блокируется. Уровень срабатывания равен 1.

2. Перенесение логарифма из сомножителей знаменателя в сомножители числителя.

   $$\forall_{abcd}\left(0 < a\ \&\ \neg(a - 1 = 0) \rightarrow \frac{b}{c\log_d a} = \frac{b\log_a d}{c}\right)$$

   Для натуральных логарифмов прием блокируется. Уровень срабатывания равен 3.

3. Деление слагаемого числителя на знаменатель.

   $$\forall_{abc}\left(\frac{ab + c}{b} = a + \frac{c}{b}\right)$$

   Уровень срабатывания равен 2.

4. Выражение частного одинаковых степеней синуса и косинуса через тангенс и котангенс.

$$\forall_{abcd}\left(\frac{a(\sin b)^c}{d(\cos b)^c}=\frac{a(\operatorname{tg} b)^c}{d}\right)$$

$$\forall_{abcd}\left(\frac{a(\cos b)^c}{d(\sin b)^c}=\frac{a(\operatorname{ctg} b)^c}{d}\right)$$

$$\forall_{abcdef}\left(\frac{d\sin((a-b)f/c)}{e\cos((b-a)f)/c}=\frac{d\operatorname{tg}((a-b)f/c)}{e}\right)$$

Уровень срабатывания равен 1.

5. Усмотрение тангенса либо котангенса двойного аргумента.

$$\forall_{abcd}\left(\frac{c(a-a(\operatorname{tg} b)^2)}{d\operatorname{tg} b}=\frac{2a\operatorname{ctg}(2b)c}{d}\right)$$

$$\forall_{abcd}\left(\frac{a\operatorname{tg} b}{d(c-c(\operatorname{tg} b)^2)}=\frac{a\operatorname{tg}(2b)}{2cd}\right)$$

6. Усмотрение числа сочетаний.

$\forall_{abmnk}(n-$натуральное & $m-$целое & $k-$целое & $0\le m$ & $0\le k$ & $n=m+k\to an!/bm!k!=aC_n^k/b)$

Указатель "дробь(фикс(0 1)фикс(0 2))" обеспечивает применение преобразования с перестановкой числителя и знаменателя.

7. Деление дроби.

$$\forall_{abc}\left(\frac{a/b}{c}=\frac{a}{bc}\right)$$

8. Сокращение для логарифма степени.

$$\forall_{abcde}\left(\frac{\log_a(b^c)d}{ce}=\frac{\log_a|b|d}{e}\right)$$

9. Выражение частного одинаковых степеней гиперболических синуса и косинуса через гиперболические тангенс и котангенс.

$$\forall_{abcd}\left(\frac{a(\operatorname{sh} b)^c}{d(\operatorname{ch} b)^c}=\frac{a(\operatorname{th} b)^c}{d}\right)$$

$$\forall_{abcd}\left(\frac{a(\operatorname{ch} b)^c}{d(\operatorname{sh} b)^c}=\frac{a(\operatorname{cth} b)^c}{d}\right)$$

10. Частичное сокращение на тангенс.

$$\forall_{abcde}\left(\neg(\sin b = 0) \to \frac{a\operatorname{tg} b}{(c\operatorname{tg} b + d)e} = \frac{a}{(c + d\operatorname{ctg} b)e}\right)$$

11. Переход к сигнуму.

$$\forall_{abc}\left(\frac{|a|b}{ac} = \frac{\mathrm{sg}ab}{c}\right)$$

$$\forall_{abc}\left(\frac{ab}{|a|c} = \frac{\mathrm{sg}ab}{c}\right)$$

$$\forall_{bdeg}\left(\frac{be|d/e|}{dg} = \frac{b\mathrm{sg}(de)}{g}\right)$$

$$\forall_{abce}\left(\neg(a = 0)\ \&\ a - \text{число} \to \frac{be}{|ce|} = \frac{b\mathrm{sg}e}{|c|}\right)$$

12. Переход к модулю.

$$\forall_{acd}\left(\frac{a\mathrm{sg}\operatorname{tg} d}{c\operatorname{tg} d} = \frac{a|\operatorname{ctg} d|}{c}\right)$$

13. Усмотрение тангенса суммы.

$\forall_{abcdef}(c-\text{натуральное}\ \&\ \neg(d-d\operatorname{tg} a\operatorname{tg} b = 0) \to e(\operatorname{tg} a+\operatorname{tg} b)^c/f(d-d\operatorname{tg} a\operatorname{tg} b)^c = e(\operatorname{tg}(a+b))^c/fd^c)$

$\forall_{abcdef}(c - \text{натуральное}\ \&\ \neg(\cos(a+b) = 0)\ \&\ \neg(d = 0) \to e(\operatorname{tg} a + \operatorname{tg} b)^c/f(d - d\operatorname{tg} a\operatorname{tg} b)^c = e(\operatorname{tg}(a+b))^c/fd^c)$

Нормализатор стандартизации дробей с неизвестными "уравндробь"

Этот нормализатор невелик. Кроме рекурсивных обращений к обработке числителя и знаменателя, он имеет единственный прием:

$$\forall_{abcde}\left(0 \le a\ \&\ 0 < b\ \&\ \neg(e = 0) \to \frac{da^c}{eb^c} = \frac{d(a/b)^c}{e}\right)$$

Здесь a, b не содержат неизвестных, c - содержит. Таким образом, выполняется группировка под общий неизвестный показатель степени двух степеней с известными основаниями. Рекурсивные обращения к обработке числителя и знаменателя имеют вид:

$$\forall_{abc}\left(c = a \to \frac{a}{b} = \frac{c}{b}\right)$$

$$\forall_{abc}\left(\neg(b = 0)\ \&\ c = b \to \frac{a}{b} = \frac{a}{c}\right)$$

В обоих случаях роль нормализатора играет оператор "нормуравн", переадресующий свои входные данные нормализатору стандартизации неизвестных выражений с соответствующим заголовком.

Нормализатор "простейшиедроби"

Нормализатор преобразует дробное выражение к виду суммы простейших дробей, знаменателями которых служат степени неприводимых многочленов относительно заданной переменной x. Она сообщается нормализатору через комментарий (переменная x). Перед обращением знаменатель дроби уже должен быть разложен на множители, что обычно обеспечивается срабатыванием изложенных выше приемов стандартизации дробей. Нормализатор имеет следующие приемы:

1. Минус перед дробью. Если перед дробью стоит минус, то выполняется рекурсивное обращение к преобразованию дроби без минуса:

   $\forall_{ab}(a = b \to -b = -a)$.

   Антецедент обрабатывает выражение b нормализатором "простейшиедроби". Преобразование реализуется, если результат a имеет вид суммы, возможно, с вынесенным перед ней знаком "минус".

2. Выделение множителей числителя и знаменателя, не зависящих от рассматриваемой переменной. Прием реализует рекурсивное обращение к обработке дроби с отброшенными такими множителями:

   $$\forall_{abcde}\left(\frac{d}{c} = e \to \frac{ad}{bc} = \frac{a}{b}e\right)$$

   Указатель "вход(переменная x6)" идентифицирует переменную x6, относительно которой берутся многочлены. Указатели "перечень(x4 входит(x6 x4))", "перечень(x3 входит(x6 x3))" определяют c, d как произведения всех сомножителей знаменателя и числителя, не содержащих данную переменную. Проверяется, что хотя бы одно из этих произведений невырожденное. Затем антецедент обращается к нормализатору "простейшиедроби". Результирующее выражение обрабатывается нормализатором раскрывания скобок "стандплюс".

3. Устранение дробного слагаемого в числителе. Прием обеспечивает стандартизацию многоэтажных дробных выражений, которые могут возникать в процессе вычислений:

   $$\forall_{abcd}\left(\frac{a + b/c}{d} = \frac{a}{d} + \frac{b}{cd}\right)$$

   Здесь знаменатель c содержит выделенную переменную x. Каждое из слагаемых правой части обрабатывается нормализаторами "нормдробь", "простейшиедроби". К полученной сумме применяется нормализатор "нормплюс".

4. Ускорение вычислений в специальном случае. Для ускорения вычислений рассмотрен один частный случай, вынесенный в отдельный прием:

$$\forall_{abcdefg}\left(c = d - b \;\&\; \neg(c = 0) \to \frac{e}{f(a + bg)(a + dg)} = \frac{e}{fcg(a + bg)} - \frac{e}{fcg(a + dg)}\right)$$

   Здесь g - выделенная переменная, a - многочлен, степень которого больше единицы. На выражения e, f, b, d ограничения не накладываются (в частности, b, d

могут содержать g). Каждая из дробей заменяющей части обрабатывается нормализаторами "нормдробь", "простейшиедроби", а вся заменяемая часть - нормализаторами "нормплюс", "стандплюс".

5. Применение рекуррентной формулы, понижающей степень знаменателя. Это основной шаг процедуры перехода к сумме простейших дробей:

   $\forall_{afghpuvw}$(частноемногочленов$(\lambda_x(f(x), x -$ число$), \lambda_x(g(x), x -$ число$), h, u)$ &
   частноемногочленов$(\lambda_x(v(x), x -$ число$), \lambda_x(g(x), x -$ число$), w, p) \to$
   $f(x)/(g(x))^a v(x) = (h(x)p(x) - u(x)w(x))/p(x)(g(x))^{a-1}v(x) + u(x)/p(x)(g(x))^a)$

   В то время, как предшествующие приемы срабатывали на уровне 1, данный прием срабатывает на уровне 2. Переменная a идентифицируется с натуральной константой, возможно, равной 1. x - выделенная переменная. Она входит в многочлены $g(x), v(x)$. Степень многочлена $v(x)$ не меньше степени $g(x)$, причем остаток $p(x)$ от деления $v(x)$ на $g(x)$ ненулевой. Чтобы отсечь случаи чрезмерно трудоемких вычислений, , наложено ограничение $a < 5$. Антецеденты выполняют деление многочленов с остатком, используя для этого пакетный синтезатор "частноемногочленов". Выходные переменные h, u, w, p получают в качестве своих значений термы вида "отображение(...)". Чтобы эти переменные можно далее было использовать как функциональные переменные, введены указатели "функция(x8 x23)", "функция(x20 x23)", "функция(x22 x23)", "функция(x15 x23)". Например, если h было идентифицировано как терм $\lambda_y(A(y), y -$ число$)$, то $h(x)$ формируется в виде $A(x)$. Заметим, что при обращении к синтезатору делимое и делитель обрабатываются нормализатором раскрывания скобок "стандплюс". Предварительно применяется нормализатор "сокращплюс", ускоряющий процесс раскрывания скобок за счет усмотрения разностей квадратов. Каждая из дробей заменяющего терма приема обрабатывается нормализаторами "нормдробь", "простейшиедроби", а весь заменяющий терм - нормализаторами "стандплюс", "нормплюс".

6. Уменьшение степени числителя.

   Если степень числителя как многочлена от выделенной переменной x не меньше степени многочлена, являющегося основанием степени сомножителя змаенателя, то числитель делится на последний многочлен с остатком:

   $\forall_{abfghu}$(частноемногочленов$(\lambda_x(f(x), x -$ число$), \lambda_x(g(x), x -$ число$), h, u) \to$
   $f(x)/b(g(x))^a = h(x)/b(g(x))^{a-1} + u(x)/b(g(x))^a)$

   Как и выше, a идентифицируется с натуральной константой. Степень многочлена $f(x)$ не меньше степени $g(x)$. Выражение b не содержит x, т.е. прием применяется лишь после того, как знаменатель становится (с точностью до коэффициента) степенью многочлена, не поддающегося разложению на множители. Многочлен $g(x)$ зависит от x. Прием срабатывает на уровне 2.

10.8 Приемы символа "степень"

Выражение "степень(a b)", обозначающее результат возведения вещественного числа a в вещественную степень b, прорисовывается формульным редактором в виде

$$a^b.$$

Считается, что выражение имеет смысл, если либо $a > 0$, либо $a = 0$ и $b > 0$, либо $a < 0$ и b - рациональное с нечетным знаменателем. Справочник "одз" для степени реализован непосредственно на ЛОСе (см. начало данной главы). Справочники "арность", "тип", "область", "единица" определяют простейшие свойства операции (имеет два операнда, принимает вещественные значения, причем любые, имеет единицу 1 для второго операнда). Справочник "общаястепень" используется компилятором - он указывает, что для идентификации наибольшего общего делителя показателей нескольких степенных выражений применяется процедура "общаястепень". Справочники "общнорм", "конст", "дистрибразвертка", "неизвоценка", "подстановка", "неизв", "упрощнеизв", "упрощтерм", "обобщподст" используются при выводе в базе теорем и автоматическом синтезе приемов. Справочник "вычисл" указывает компилятору операторы ЛОСа, используемые при вычислениях со степенями.

Общая стандартизация выражений

Для общей стандартизации степенных выражений используются следующие приемы:

1. Возведение десятичного числа в натуральную степень.

   $\forall_{abc}(c = a^b \rightarrow a^b = c)$

   a - десятичная константа, b - натуральная константа. Антецедент обращается к процедуре "натурстепень", выполняющей вычисления. Чтобы в задаче не появлялись чрезмерно большие числа, введены следующее ограничения. Если текущая задача - на преобразование и имеет цель "число" (т.е. требуется провести вычисления), то b не превосходит 1000. Иначе - при $a = 2$ должно выполняться $b \leq 27$, при $a = 3$ - $b \leq 12$, в остальных случаях $b \leq 8$. Если a равно 2 либо 5, причем данное степенное выражение умножено на другое степенное выражение с основанием 2 либо 5, то прием блокируется. Это позволяет сразу усматривать степени десяти. Уровень срабатывания равен 0.

2. Основание степени единица.

   $\forall_a(1^a = 1)$

   Этот и два следующих приема имеют уровень срабатывания 0.

3. Основание степени ноль.

   $\forall_a(0^a = 0)$

4. Показатель степени единица.

   $\forall_a(a^1 = a)$

5. Показатель степени ноль.

   $\forall_a(a^0 = 1)$

   Уровень срабатывания равен 1. Так как в о.д.з. случай $a = 0$ невозможен, то конфликты с приемом "основание степени ноль" не возникают. При разборе случаев иногда могут возникать выражения вида 0^0, которые будут заменяться на 0. Однако, список посылок здесь будет ложным, и эти замены несущественны. Вводить проверку условия $\neg(a = 0)$ в данном приеме и условия $a > 0$ в приеме "основание степени ноль" нежелательно из-за неоправданного замедления вычислений. Это вполне согласуется с принципом отбрасывания проверки тех условий, которые вытекают из требований на о.д.з. преобразуемых выражений.

6. Минус в основании степени.

   $\forall_{ab}(b-\text{rational}\ \&\ \neg(\text{числитель}(b)-\text{even})\ \&\ \neg(\text{знаменатель}(b)-\text{even})\rightarrow(-a)^b=-a^b)$

   Прием блокируется, если выражение относится к уравнению задачи на описание, в котором $-a$ находится также под радикалом четной степени.

   $\forall_{ab}(b-\text{rational}\ \&\ \text{числитель}(b)-\text{even}\rightarrow(-a)^b=a^b)$

   Прием применяется без ограничений. Уровни срабатывания в обоих случаях равны 0.

7. Минус в показателе степени.

   $$\forall_{ab}\left(a^{-b}=\frac{1}{a^b}\right)$$

   Преобразование блокируется в следующих случаях:

   (a) a имеет вид дроби (тогда применяется другой прием, см. ниже)

   (b) Степень расположена под описателем "отображение", связывающим некоторые ее переменные.

   (c) Не решается задача на разложение в ряд Тейлора.

   (d) Если решается дифференциальное уравнение, то b не содержит ни неизвестной функции, ни варьируемой переменной.

   (e) Если решается задача на упрощение ответа, то a не равно 10.

   Уровень срабатывания приема равен 0. Создана еще одна версия приема с той же теоремой. Она срабатывает на уровне 5 при решении дифференциальных уравнений, если b содержит неизвестную функцию. При этом сохраняются первые три фильтра предыдущего приема.

   $$\forall_{abc}\left(\left(\frac{a}{b}\right)^c=\left(\frac{b}{a}\right)^c\right)$$

   Уровень срабатывания равен 0.

8. Произведение в основании степени.

   $\forall_{abc}(0\le a\ \&\ 0\le b\rightarrow a^cb^c=(ab)^c)$

   Преобразование применяется справа налево. Блокируется оно в следующих сравнительно редких случаях:

   (a) Степень входит в условие задачи на преобразование, имеющей комментарий "длина". Этот комментарий указывает на этап завершающей компактной переформулировки терма.

   (b) Степень расположена в условии задачи на описание, решаемой для редактирования ответа (кроме редактирования теорем).

(c) Степень расположена под описателем, связывающая приставка которого пересекается с параметрами показателя c и не пересекается с параметрами основания ab.

(d) Степень расположена в условии задачи, причем показатель степени содержит неизвестные, а основание - не содержит.

(e) Степень входит в условие задачи, имеющей цель "свертка".

Если основание степени имеет ровно два сомножителя, то a идентифицируется с тем из них, который идет первым в лексикографическом порядке. Таким образом блокируется рассмотрение симметричных ситуаций. Кроме того, отбрасывается случай, когда показатель степени - простая дробь с нечетным знаменателем. Для него предусмотрен отдельный прием:

$$\forall_{abc}\,(c - \text{rational} \;\&\; \neg(\text{знаменатель}(c) - \text{even} \rightarrow a^c b^c = (ab)^c)$$

Ограничения на срабатывание - те же, что и выше. Уровни срабатывания обоих приемов равны 0.

Вернемся к теореме первого приема. Иногда разбиение основания на два неотрицательных множителя непосредственно не усматривается, однако может быть подсказано неравенством из посылок. Например, если в посылках имеется неравенство $0 \leq ab$, то основание степени ab^3 можно разбить на два неотрицательных сомножителя ab, b^2. Для таких случаев по первой теореме создана еще одна версия приема. У нее второй антецедент не выделен указателем "блокпроверок", а идентифицируется непосредственно с посылкой. При этом требуется, чтобы остаточное произведение a имело не менее двух сомножителей. Уровень срабатывания равен 0.

Если основание степени представляет собой произведение двух неположительных сомножителей, то применяется следующий прием:

$$\forall_{abc}\,(a \leq 0 \;\&\; b \leq 0 \rightarrow (-a)^c(-b)^c = (ab)^c)$$

Отличие от первого приема состоит лишь в том, что уровень срабатывания равен 1. Разбиение в произведение двух неотрицательных сомножителей оказывается возможным существенно чаще, и за счет увеличения уровня срабатывания для неположительных сомножителей достигается экономия трудоемкости.

В особых случаях проверяется неотрицательность лишь одного сомножителя. Это допустимо, если сопровождение по о.д.з. большой роли не играет (задачи на исследование с целью "известно" и решение функциональных уравнений):

$$\forall_{abc}\,(0 \leq a \rightarrow a^c b^c = (ab)^c)$$

Уровень срабатывания здесь равен 0.

При решении дифференциальных уравнений неотрицательность сомножителей основания не обязательна, так как степень произведения преобразуется в произведение степеней модулей:

$$\forall_{abc}\,((ab)^c = |a|^c|b|^c)$$

$$\forall_{abc}\,(|ab|^c = |a|^c|b|^c)$$

Переменная c идентифицируется с простой дробью, имеющей четный знаменатель. Выражение a содержит варьируемую переменную, b - неизвестную функцию. В первом случае условие задачи не должно содержать производную. Уровни срабатывания равны 1.

9. Дробь в основании степени.

$$\forall_{abc}\left(c - \text{rational} \ \& \ \neg(\text{знаменатель}(c) - \text{even}) \to \left(\frac{a}{b}\right)^c = \frac{a^c}{b^c}\right)$$

Преобразование блокируется в следующих редких случаях:

(a) Степень входит в условие задачи на преобразование, имеющей комментарий "длина".

(b) Степень расположена в условии задачи на описание, решаемой для редактирования ответа (кроме редактирования теорем).

(c) Степень с неизвестными расположена в условии задачи на описание, имеющей также вхождение выражения $|a/b|$ либо $|b/a|$.

(d) Степень расположена под описателем "отображение", связанная переменная которого входит в c и не входит в a, b.

(e) Степень расположена под описателем "отображение" и содержит его связанную переменную, причем задача имеет цель "исследовать". Такие задачи используются для общей характеризации объектов (например, при исследовании свойств функции вещественной переменной).

(f) Степень находится в условии задачи на преобразование, решаемой для упрощения подынтегрального выражения. При этом показатель степени c - дробный, а основание степени имеет вид $(px+q)/(rx+s)$, где x - переменная интегрирования.

(g) Степень расположена в посылке задачи, причем внутри описателя "отображение".

(h) Степень находится в условии задачи, имеющей цель "свертка".

(i) Степень находится в условии задачи, показатель степени содержит неизвестные, основание - не содержит, причем числитель a отличен от единицы.

Уровень срабатывания равен 0.

$$\forall_{abc}\left(0 \leq a \ \& \ 0 < b \to \left(\frac{a}{b}\right)^c = \frac{a^c}{b^c}\right)$$

Этот прием блокируется в следующих случаях:

(a) Степень входит в условие задачи на преобразование, имеющей комментарий "длина".

(b) Степень расположена в условии задачи на описание, решаемой для редактирования ответа (кроме редактирования теорем).

(c) Степень расположена в условии задачи на преобразование, упрощающей выражение, анализируемое на убывание в точке.

(d) Показатель степени - простая дробь с нечетным знаменателем (тогда применяется предыдущий прием).

(e) Степень находится в условии задачи, показатель степени содержит неизвестные, основание - не содержит, причем числитель a отличен от единицы.

(f) Степень находится в условии задачи, имеющей цель "свертка".

(g) Степень расположена под описателем "отображение", связанная переменная которого входит в c.

(h) Степень находится в условии задачи на преобразование, решаемой для упрощения подынтегрального выражения. При этом показатель степени c - дробный, а основание степени имеет вид $(px+q)/(rx+s)$, где x - переменная интегрирования.

(i) Степень находится в условии задачи на преобразование, решаемой для упрощения подынтегрального выражения. Переменная интегрирования входит в c.

(j) Степень находится в условии задачи на преобразование, решаемой для упрощения подынтегрального выражения. Переменная интегрирования входит в a/b, причем каждое вхождение переменной интегрирования в это выражение является основанием степени с дробным показателем.

Прием имеет два уровня срабатывания - 0 и 2. Если на уровне 0 истинность антецедентов не усматривается, то на уровне 2 делается повторная попытка усмотрения.

Еще один случай - неположительные числитель и знаменатель:

$$\forall_{abc}\left(a \leq 0 \;\&\; b < 0 \to \left(\frac{a}{b}\right)^c = \frac{(-a)^c}{(-b)^c}\right)$$

Фильтры аналогичны предыдущему случаю, уровень срабатывания равен 0.

Если неотрицательность либо неположительность операндов дроби не усматривается, но удается усмотреть знак произведения части сомножителей числителя и знаменателя, то применяются следующие приемы:

$$\forall_{abcd}\left(0 \leq a \to \left(\frac{ab}{c}\right)^d = a^d\left(\frac{b}{c}\right)^d\right)$$

$$\forall_{abceg}\left(0 \leq ab \;\&\; 0 \leq eg \to \left(\frac{a}{b}\right)^c\left(\frac{g}{e}\right)^c = \left(\frac{ag}{be}\right)^c\right)$$

$$\forall_{abceg}\left(ab \leq 0 \;\&\; eg \leq 0 \to \left(-\frac{a}{b}\right)^c\left(-\frac{g}{e}\right)^c = \left(\frac{ag}{be}\right)^c\right)$$

Первый прием срабатывает на уровне 2. Два последних приема выполняют замену справа налево и срабатывают на уровне 3. Переменные a, b идентифицируются у них как отдельные сомножители числителя и знаменателя. Затем проверяются антецеденты.

Преобразование степени дроби в частное степеней выполняется при выводе теоремы, если данная степень является сомножителем числителя:

$$\forall_{abcdn}\left(\frac{a(b/c)^n}{d}=\frac{ab^n}{dc^n}\right)$$

Текущая задача имеет тип "преобразовать" и комментарии "сокращение", "длина". n - натуральная константа. Уровень срабатывания равен 3.

10. Повторное возведение в степень.

    $\forall_{abc}(b-\text{rational}\ \&\ c-\text{rational}\ \&\ \neg(\text{знаменатель}(b)-\text{even})\ \&\ \neg(\text{знаменатель}(c)-\text{even})\to(a^b)^c=a^{bc})$

    $\forall_{abc}(0\le a\to(a^b)^c=a^{bc})$

    $\forall_{abc}(\neg(\text{знаменатель}(b)-\text{even})\ \&\ \neg(\text{числитель}(b)-\text{even})\ \&\ b-\text{rational}\ \&\ a\le 0\to((-a)^b)^c=(-a)^{bc})$

    $\forall_{abc}(b-\text{rational}\ \&\ \text{числитель}(b)-\text{even}\to|a|^{bc}=(a^b)^c)$

    Первый прием срабатывает на уровне 0, второй и третий - на уровнях 0 и 3. Последний прием выполняет замену справа налево и срабатывает на уровне 1.

11. Умножение степеней с одинаковыми основаниями.

    $\forall_{abc}(0\le a\to a^ba^c=a^{b+c})$

    Преобразование блокируется, если целочисленная степень десятичной константы умножается на радикал (например, $2\sqrt{2}$). Оно также блокируется при формальном интегрировании - если a содержит переменную интегрирования, причем одна степень представляет собой радикал, а другая имеет целочисленный показатель. Уровни срабатывания равны 0 и 3.

    $\forall_{abc}(b-\text{rational}\ \&\ c-\text{rational}\ \&\ \neg(\text{знаменатель}(b)-\text{even})\ \&\ \neg(\text{знаменатель}(c)-\text{even})\to a^ba^c=a^{b+c})$

    Фильтры аналогичны предыдущему приему, уровень срабатывания равен 0. Хотя уровень срабатывания 0 у обоих приемов общий, сначала предпринимается попытка применения второго приема, а затем - первого. Это обеспечивается подходящей очередностью компиляции (сначала компилируется второй прием, затем первый) и размещением программ приемов в блоке программ.

    $\forall_{abc}(\text{числитель}(b)-\text{even}\ \&\ b-\text{rational}\to a^b|a|^c=|a|^{b+c})$

    Прием срабатывает без ограничений на уровне 0.

    В заключение приведем прием, где одна из степеней "спрятана" внутри натуральной константы:

    $\forall_{abcmn}(b=ac\to ba^{n-m}=ca^{n-m+1})$

    Переменные a,b,m идентифицируются с натуральными константами; n - неконстантное выражение. Антецедент выделен указателем "программа". Он предпринимает попытку деления b на a. В случае успеха коэффициент заменяется на результат деления, а показатель степени при a увеличивается на единицу. Уровень срабатывания равен 5.

12. Умножение степеней с основаниями, отличающимися знаком.

    $\forall_{abc}(\neg(\text{знаменатель}(a)-\text{even})\ \&\ \text{числитель}(c)-\text{even}\ \&\ a-\text{rational}\ \&\ c-\text{rational}\to b^c(-b)^a=(-b)^{a+c})$

$\forall_{abc}(\neg(\text{знаменатель}(a) - \text{even})\ \&\ \neg(\text{знаменатель}(c) - \text{even})\ \&\ \neg(\text{числитель}(c) - \text{even})\ \&\ a - \text{rational}\ \&\ c - \text{rational} \rightarrow b^c(-b)^a = -(-b)^{a+c})$

$\forall_{cdef}(\neg(\text{знаменатель}(c) - \text{even}\ \&\ \neg(\text{знаменатель}(f) - \text{even})\ \&\ c - \text{rational}\ \&\ f - \text{rational} \rightarrow (d-e)^c(e-d)^f = (d-e)^{c+f}(-1)^f)$

$\forall_{cdef}(\neg(\text{знаменатель}(f) - \text{even})\ \&\ f - \text{rational}\ \&\ 0 \le d - e \rightarrow (d-e)^c(e-d)^f = (d-e)^{c+f}(-1)^f)$

Уровни срабатывания равны 1. Последний прием блокируется, если преобразуется условие задачи, причем f содержит неизвестные.

13. Умножение степени произведения на произведение одинаковых степеней сомножителей основания.

    $\forall_{abcd}(0 \le ab\ \&\ c - \text{rational}\ \&\ \neg(\text{знаменатель}(c) - \text{even}) \rightarrow a^c b^c (ab)^d = (ab)^{c+d})$

    Уровень срабатывания равен 1.

14. Деление степеней с одинаковыми основаниями.

    $\forall_{abcde}(0 < a \rightarrow da^b/ea^c = da^{b-c}/e)$

    Срабатывание блокируется, если a - десятичная константа, $c = 1$, b - числовая дробь (например, $\sqrt{2}/2$). Отсекаются случаи срабатывания, охватываемые другими приемами ($b = c = 1$ либо случай рациональных показателей степени, имеющих нечетные знаменатели). Уровни срабатывания равны 1 и 3.

    $\forall_{abcde}(b - \text{rational}\ \&\ c - \text{rational}\ \&\ \neg(\text{знаменатель}(b) - \text{even})\ \&\ \neg(\text{знаменатель}(c) - \text{even}) \rightarrow da^b/ea^c = da^{b-c}/e)$

    Уровень срабатывания равен 1.

15. Деление степеней с основаниями, отличающимися знаком.

    $\forall_{abcde}(b - \text{rational}\ \&\ \neg(\text{знаменатель}(b) - \text{even})\ \&\ a < 0 \rightarrow a^b c/(-a)^d e = (-1)^b(-a)^{b-d}c/e)$

    Уровни срабатывания равны 1 и 3. Указатель "дробь" предусматривает возможность перестановки числителя и знаменателя.

16. Сокращение при перемножении двух степеней с дробными основаниями и одинаковыми показателями.

    $$\forall_{abcde}\left(\left(\frac{a}{bc}\right)^d \left(\frac{be}{f}\right)^d = \left(\frac{ae}{cf}\right)^d\right)$$

    Уровень срабатывания равен 2.

17. Сокращение численных оснований степеней с одинаковыми показателями, являющихся множителями числителя и знаменателя дроби.

    $$\forall_{abcdef}\left(0 < b\ \&\ 0 \le a\ \&\ f = \frac{a}{b} \rightarrow \frac{ea^c}{db^c} = \frac{ef^c}{d}\right)$$

Переменные a, b идентифицируются с десятичными константами. Первые два антецедента выделены указателем "программа" - они непосредственно проверяют знак числовых констант. Последний антецедент обращается к нормализатору "сокращдроби" для сокращения дроби a/b. Указатель "различны(сокращдроби дробь(x1 x2))" обеспечивает проверку того, что дробь удалось сократить. Уровень срабатывания равен 0.

$$\forall_{abcdefgmn}\left(0 < a \ \& \ 0 < b \ \& \ \neg(d = 0) \to \frac{ea^{cn+f}}{db^{cm+g}} = \frac{e(a^n/b^m)^c a^f}{db^g}\right)$$

Как и выше, основания степени a, b идентифицируются с десятичными константами. Показатели степени различны, но имеют подобные слагаемые cn, cm, где n, m - целочисленные константы, c - неконстантное выражение. Происходит группировка под показатель степени c константной дроби a^n/b^m, обрабатываемой нормализаторами общей стандартизации. Указатель "различны(сокращдроби дробь(x1 x2))" обеспечивает предварительную проверку того, что a/b сократимо. Уровень срабатывания равен 0.

18. Умножение степеней с натуральными основаниями, делящимися друг на друга.

    $\forall_{abmnk}(m = nk \to m^a n^b = k^a n^{a+b})$

    Переменные m, n идентифицируются с натуральными константами. Антецедент, выделенный указателем "программа", проверяет делимость m на n. Прием блокируется, если один из показателей степени - целое число, а другой - дробь (например, $2\sqrt{6}$). Уровень срабатывания равен 3.

19. Приведение подобных членов для степеней.

    $\forall_{abcdpqrskmn}(b = md \ \& \ c = nd \ \& \ r = p - q \ \& \ s = ma^r + n \ \& \ q < p \to ba^{k+p} + ca^{k+q} = sda^{k+q})$

    Переменные a, p, q идентифицируются с десятичными константами. Первый и второй антецеденты, выделенные указателем "идентификатор", определяют общу часть d коэффициентов b, c. Проверяется, что множители m, n суть десятичные константы. Антецеденты начиная с третьего выделены указателем "программа" и выполняют арифметические вычисления, необходимые для получения нового коэффициента s. После определения показателя степени r проверяется, что он целочисленный и не превосходит 3. Уровень срабатывания равен 1.

    $\forall_{abcdkmnpq}(b = md \ \& \ c = nd \to ba^{k+p} + ca^{k+q} = (ma^p + na^q)da^k)$

    Этот прием аналогичен предыдущему, но подобные члены приводятся относительно выражений с неизвестными. Переменные a, p, q идентифицируются с выражениями, не содержащими неизвестных, переменная k - с неизвестным выражением. Первый и второй антецеденты определяют множители m, n коэффициентов b, c, получающиеся после отбрасывания их общей части d. Проверяется, что m, n не содержат неизвестных. Уровень срабатывания равен 1.

    $\forall_{abcdn}(n - \text{rational} \ \& \ \neg(\text{числитель}(n) - \text{even}) \to a(b-c)^n + d(c-b)^n = (a-d)(b-c)^n)$

    $\forall_{abcdn}(n - \text{rational} \ \& \ \text{числитель}(n) - \text{even} \to a(b-c)^n + d(c-b)^n = (a+d)(b-c)^n)$

    Переменные a, d идентифицируются с десятичными константами. Уровень срабатывания равен 1.

20. Деление на степень дроби.

$$\forall_{bcdef}\left(\frac{b}{f(c/e)^d}=\frac{b(e/c)^d}{f}\right)$$

При преобразовании подынтегральных выражений прием блокируется. Уровень срабатывания равен 2.

21. Выделение степени десятки.

$\forall_{mn}(k=\min(m,n)\rightarrow 2^m5^n=(10)^k2^{m-k}5^{n-k})$

Переменные m,n идентифицируются с натуральными константами, хотя бы одна из которых более единицы. Антецедент выделен указателем "программа". Прием срабатывает на уровне 0.

22. Произведение степеней модулей.

$\forall_{abc}(|ab|^c=|a|^c|b|^c)$

Преобразование применяется справа налево. Он блокируется при преобразовании подынтегрального выражения, а также при решении дифференциальных уравнений. Уровень срабатывания равен 1.

23. Возведение в квадрат линейной комбинации радикалов с взаимно обратными дробями.

$$\forall_{abcd}\left(\left(a\sqrt{\frac{b}{c}}+d\sqrt{\frac{c}{b}}\right)^2=\frac{a^2b^2+d^2c^2+2abcd}{bc}\right)$$

Прием применяется в условии задачи на преобразование. Уровень срабатывания равен 2.

24. Приведение подобных членов для радикалов с взаимно обратными дробями.

$$\forall_{abcd}\left(c\sqrt{\frac{a}{b}}+d\sqrt{\frac{b}{a}}=\frac{(ac+bd)\sqrt{\frac{a}{b}}}{a}\right)$$

Прием применяется в условии задачи на преобразование. Уровень срабатывания равен 3.

25. Специальные суммы двойных радикалов.

$\forall_{ab}(0\le a^2+b\;\&\;0\le a-\sqrt{a^2+b}\rightarrow\sqrt{a+\sqrt{a^2+b}}+\sqrt{a-\sqrt{a^2+b}}=\sqrt{2(a+\sqrt{-b})})$

$\forall_{ab}(0\le a^2+b\;\&\;0\le a-\sqrt{a^2+b}\rightarrow\sqrt{a+\sqrt{a^2+b}}-\sqrt{a-\sqrt{a^2+b}}=\sqrt{2(a-\sqrt{-b})})$

Антецеденты обрабатываются проверочными операторами. Указатель "посылка(и(меньшеилиравно(x2 0)меньшеилиравно(0 x1)))" обеспечивает вывод сопровождающих преобразование неравенств $b\le 0$, $0\le a$. Уровень срабатывания равен 3.

26. Выражения с числовыми радикалами.

$$\forall_{abcde}\left(\frac{(e\sqrt[4]{a}+\sqrt{b})c}{d\sqrt[4]{a}}=\frac{\left(ae+\sqrt{ba^{3/2}}\right)c}{ad}\right)$$

Выражение a - десятичная константа, выражение b не имеет переменных. Уровень срабатывания равен 1.

$\forall_{abcdn}(d=(a\sqrt{b}+c)^n\rightarrow(a\sqrt{b}+c)^n=d)$

a,b,c - десятичные константы, n - натуральная константа, не превосходящая 4. Антецедент выполняет раскрывание скобок с помощью оператора "стандплюс". Если степень сама является множителем основания степенного выражения, то прием блокируется. Уровень срабатывания равен 2.

$\forall_{abcd}((a\sqrt{b}+c\sqrt{d})^2=a^2b+c^2d+2ac\sqrt{bd})$

a,b,c,d - десятичные константы. Уровень срабатывания равен 2. Заменяющий терм обрабатывается нормализаторами общей стандартизации, выполняющими арифметические вычисления.

$\forall_{abc}((a\sqrt{b}+c)^2=a^2b+c^2+2ac\sqrt{b})$

a,b,c - десятичные константы. Преобразуемое выражение не является сомножителем основания внешней степени либо операндом внешнего логарифма. Уровень срабатывания равен 0.

$$\forall_{abcd}(n=2m\rightarrow(a\sqrt{b}+c\sqrt{d})^n=(a^2b+c^2d+2ac\sqrt{bd})^m)$$

a,b,c,d - десятичные константы. n - натуральная константа, меньшая 8. Антецедент выделен указателем "программа".

$$\forall_{abcd}\left(a\sqrt{b}+\frac{c}{\sqrt{bd}}=\frac{abd+c}{\sqrt{bd}}\right)$$

a,b,c,d - числовые константы. Уровень срабатывания равен 2.

$$\forall_{mn}(\sqrt{m}\sqrt{n}=\sqrt{mn})$$

m,n - десятичные константы. Уровень срабатывания равен 1.

$$\forall_{abc}\left(\sqrt[4]{c}\sqrt{a+b\sqrt{c}}=\sqrt{a\sqrt{c}+bc}\right)$$

a,b,c - десятичные константы. Уровень срабатывания равен 3.

$\forall_{abcde}(0\le\sqrt{b}+c\ \&\ f=(d\sqrt{b}+e)(a\sqrt{b}+c)^2\rightarrow(a\sqrt{b}+c)\sqrt{d\sqrt{b}+e}=\sqrt{f})$

$\forall_{abcde}(\sqrt{b}+c\le 0\ \&\ f=(d\sqrt{b}+e)(a\sqrt{b}+c)^2\rightarrow(a\sqrt{b}+c)\sqrt{d\sqrt{b}+e}=-\sqrt{f})$

a,b,c,d,e - десятичные константы. Второй антецедент обращается к нормализатору "стандплюс" для раскрывания скобок. Уровень срабатывания равен 3.

$$\forall_{abcd}(c=2d\ \&\ a-b=d^2\rightarrow\sqrt{a-\sqrt{b}c}=|d-\sqrt{b}|)$$

$$\forall_{abcd}(c = 2d \ \& \ a - b = d^2 \rightarrow \sqrt{a + \sqrt{b}c} = |d + \sqrt{b}|)$$

a, b, c, d - целочисленные константы. Антецеденты выделены указателем "программа". Уровень срабатывания равен 0.

$\forall_{abcdef}(c = be \ \& \ e = 2f \ \& \ a - b = f^2 \rightarrow \sqrt{d(a\sqrt{b} + c)} = \sqrt[4]{b}\sqrt{d}|f + \sqrt{b}|)$

a, b, c, d, e - целочисленные константы. Антецеденты выделены указателем "программа". Уровень срабатывания равен 0.

$$\forall_{abcm}(cm^2 + 1 = a \ \& \ b = 2m \rightarrow \sqrt{a - b\sqrt{c}} = |m\sqrt{c} - 1|)$$

$$\forall_{abcm}(cm^2 + 1 = a \ \& \ b = 2m \rightarrow \sqrt{a + b\sqrt{c}} = |m\sqrt{c} + 1|)$$

a, b, c - целочисленные константы. Антецеденты выделены указателем "программа". Второй из них проверяет четность b и делит его на 2, затем первый антецедент вычисляет $cm^2 + 1$ и сравнивает с a. Уровень срабатывания равен 0.

$$\forall_{abcdepq}\left(c = 2d \ \& \ a - b = d^2 \rightarrow \frac{e\log_p(a + \sqrt{b}c)}{2q} = \frac{e\log_p|d + \sqrt{b}|}{q}\right)$$

a, b, c - целочисленные константы. Антецеденты выделены указателем "программа. Уровень срабатывания равен 0.

$$\forall_{abcdep}\left(p = a^2 - b^2c \ \& \ \neg(p = 0) \rightarrow \frac{\sqrt{a + b\sqrt{c}}d}{\sqrt{a - b\sqrt{c}}e} = \frac{(a + b\sqrt{c})d}{\sqrt{p}e}\right)$$

a, b, c - десятичные константы. Антецеденты выделены указателем "программа". Уровень срабатывания равен 2.

$$\forall_{abcd}((a + b\sqrt{c})\sqrt{d} = a\sqrt{d} + b\sqrt{cd})$$

a, b, c, d - десятичные константы. Преобразуемое выражение является слагаемым суммы, находящейся под внешним радикалом. Уровень срабатывания равен 2.

27. Специальные случаи упрощения выражений с радикалами.

$$\forall_{ab}\left(a(b + 1)^{3/2} - a\sqrt{b + 1} = ab\sqrt{b + 1}\right)$$

Уровень срабатывания равен 1.

$\forall_{ab}(a - b^2 = 0 \rightarrow \sqrt{a} = |b|)$

Радикал встречается в условии задачи на описание; антецедент идентифицируется с утверждением из контекста замены, причем выражение b не содержит неизвестных. Уровень срабатывания равен 2.

$$\forall_{abcd}\left(\frac{a\sqrt{\sqrt{b^2+c}-b}}{d(\sqrt{\sqrt{b^2+c}+b}+\sqrt{\sqrt{b^2+c}-b})}=\frac{a\sqrt{c}}{d(\sqrt{b^2+c}+b+\sqrt{c})}\right)$$

Уровень срабатывания равен 1.

$$\forall_{abc}\left(c\sqrt{\frac{a}{c}}\sqrt{\frac{b}{c}}=\mathrm{sg}c\sqrt{ab}\right)$$

Уровень срабатывания равен 2.

$\forall_{abc}(b=a-c^2\ \&\ 0\le c\to\sqrt{a-b}=c)$

Радикал входит в условие задачи на преобразование, причем антецедент идентифицируется с утверждением из контекста замены. Уровень срабатывания равен 3.

28. Сложение дробей под радикалом при интегрировании. Заметим, что сложение дробных выражений под радикалом не относится к числу преобразований общей стандартизации. В некоторых случаях (например, при решении уравнений, где радикалы устраняются путем возведения в квадрат) оно может оказаться избыточным. Однако, выделен ряд ситуаций, когда данное преобразование применяется.

$$\forall_{abc}\left(0<c\to\sqrt{a+\frac{b}{c}}=\frac{\sqrt{ac+b}}{\sqrt{c}}\right)$$

Радикал входит в условие задачи на преобразование подынтегрального выражения. Переменная интегрирования не встречается в c. Прием срабатывает на уровне 4.

29. Сложение дробей под радикалом в посылках задачи на исследование.

$$\forall_{abcd}\left(d=\frac{a}{b}+c\to\sqrt{\frac{a}{b}+c}=\sqrt{d}\right)$$

Прием применяется в задачах на исследование, имеющих цель "известно". Текущая посылка не должна иметь числовых атомов, отличных от переменных. При этом радикал должен являться сомножителем знаменателя дроби. Уровень срабатывания равен 4.

30. Группировка радикалов, при которой коэффициент оказывается равен выражению под радикалом.

$$\forall_{abc}\left(ca\sqrt{a+b}+cb\sqrt{a+b}=c(a+b)^{3/2}\right)$$

$$\forall_{abc}\left(ca\sqrt{a-b}-cb\sqrt{a-b}=c(a-b)^{3/2}\right)$$

Прием применяется в условии задачи на преобразование. Уровень срабатывания равен 3.

31. Изменение основания степени, содержащего целочисленный радикал. Попытка представить константное выражение $m+n\sqrt{b}$, расположенное в основании степени, как натуральную степень другого константного основания степени $c+a\sqrt{b}$:

    $\forall_{abcmnpx}(a^2+c^2 < m^2+n^2$ & радикстепень$(c+a\sqrt{b}, m+n\sqrt{b}, p) \to (m+n\sqrt{b})^x = p^x)$

    Здесь m, n - целочисленные константы, b - натуральная константа. Преобразование выполняется в условии задачи на описание, причем показатель степени x содержит неизвестные. Указатель "контекст(позиция(x6 корень)вид(x6 степень(плюс(умножение(x1 степень(x2 дробь(1 2)))x3)x24))не(известно(x24)) единица(1 x1)заменазнака(минус x1))" позволяет идентифицировать в том же самом условии подвыражение $(c+a\sqrt{b})^y$, где a, c - целочисленные константы, y не известно. Проверяется, что $a \neq n$ либо $c \neq m$. Первый антецедент выделен указателем "программа", второй - обрабатывается пакетным синтезатором "радикстепень". Этот синтезатор выполняет последовательные деления константы $m+n\sqrt{b}$ на $c+a\sqrt{b}$, пока не будет получена единица. В результате возникает выражение p вида $(c+a\sqrt{b})^k$. Программа синтезатора может быть найдена в корневом подразделе раздела "степени". Прием срабатывает на уровне 6.

32. Вынесение из-под радикала полного квадрата.

    $$\forall_{abcd}\left(0 \le a \to \sqrt{\frac{a^2b}{c}+\frac{a^2d}{e}} = a\sqrt{\frac{b}{c}+\frac{d}{e}}\right)$$

    Прием применяется в задачах на исследование, имеющих цель "известно". Уровень срабатывания равен 1.

    $$\forall_{abc}\left(\sqrt{\frac{a^2b}{c}} = |a|\sqrt{\frac{b}{c}}\right)$$

    Прием применяется в условии задачи на преобразование. a не должно иметь вида степени с дробным показателем. Уровень срабатывания равен 2.

    $$\forall_{abc}\left(\sqrt{\frac{b}{a^2c}} = \frac{\sqrt{b/c}}{|a|}\right)$$

    Аналогично предыдущему, но уровень срабатывания равен 4.

33. Усмотрение нулевой разности радикалов.

    $\forall_{abcd}(a^2b - c^2d = 0 \to a\sqrt{b} - c\sqrt{d} = 0)$

    Прием применяется в задачах на исследование, имеющих цель "известно". Антецедент использует нормализатор раскрывания скобок "стандплюс". Выражения b, d имеют своим заголовком символ "плюс". Либо a, либо c отлично от единицы. Уровень срабатывания равен 5.

34. Преобразование дробного слагаемого показателя степени к виду суммы дробей. Общая стандартизация показателей степени предполагает преобразование их к виду сумм. Это обеспечивается следующим приемом:

$$\forall_{abcdefgh}\left(h=\frac{(b+c)^g e}{f}+d\rightarrow a^{(b+c)^g e/f+d}=a^h\right)$$

Преобразуемая степень входит либо в условие задачи на преобразование, либо в условие задачи на описание. В первом случае либо выражение e не должно содержать неизвестных, либо выражение $b+c$ должно их содержать. Во втором случае выражение $b+c$ должно содержать неизвестные. Переменная g идентифицируется с натуральной константой, меньшей 4. Антецедент последовательно обращается к нормализаторам "видумножение" и "стандплюс". Первое обращение бывает необходимо для сокращения дроби. Допускаются вырожденные единичные значения переменных e, f, g, а также остутствие слагаемого d. Прием срабатывает на уровне 1. Заметим, что встречные преобразования, нарушающие данную стандартизацию, следует вводить осторожно, чтобы избежать зацикливания. Они должны предварительно анализировать внешний контекст, блокируя случаи показателей степени.

35. Стандартизация основания степени путем одновременного изменения знака у всех его слагаемых. Применяется к степеням, имеющим рациональный показатель с нечетным знаменателем:

    $\forall_{abc}((a-b)^c=(b-a)^c)$

    $\forall_{abc}((a-b)^c=-(b-a)^c)$

    Показатель степени имеет заголовок "дробь". В первом случае переменная c идентифицируется с простой дробью, имеющей четный числитель и нечетный знаменатель, во втором - с простой дробью, имеющей нечетные числитель и знаменатель. В обоих случаях заменяющая сумма $b-a$ лексикографически предшествует заменяемой. Прием применяется в условии задачи на преобразование либо описание. Во втором случае блокируется его применение под знаком производной. Заметим, что в обоих случаях число слагаемых в основании степени может быть любым - требуется лишь, чтобы одно из них начиналось с минуса. Нормализаторы общей стандартизации обеспечивают коррекцию знаков у слагаемых суммы $b-a$. Уровень срабатывания равен 2. Для первой теоремы предусмотрена еще одна версия приема, срабатывающая на уровне 5 в посылках задач на доказательство либо исследование. При этом уже должна иметься посылка, содержащая выражение $(b-a)^c$. В геометрических задачах эта версия блокируется.

36. Минус единица в основании степени.

    $\forall_{ai}((-1)^{a+i}=(-1)^i)$

    a - целочисленная четная константа. Уровень срабатывания равен 0.

    $\forall_{ai}((-1)^{a+i}=(-1)^{i+1})$

    a - целочисленная нечетная константа, отличная от 1. Уровень срабатывания равен 0.

    $\forall_a(-(-1)^a=(-1)^{a+1})$

    Замена выполняется справа налево на уровне 0.

$$\forall_{mnp}\left(\neg(m-\text{even})\ \&\ n-\text{целое}\ \&\ p-\text{целое}\rightarrow(-1)^{mn/p}=(-1)^{n/p}\right)$$

m - целочисленная константа. Уровень срабатывания равен 1.

$\forall_n(n - \text{целое} \mathbin{\&} \neg((-1)^n - 1 = 0) \rightarrow (-1)^n = -1)$

$\forall_n(n - \text{целое} \mathbin{\&} \neg((-1)^n + 1 = 0) \rightarrow (-1)^n = 1)$

Первый антецедент обрабатывается проверочным оператором, второй непосредственно берется из контекста. Уровень срабатывания равен 0.

$\forall_{ai}(a - \text{even} \rightarrow (-1)^{a+i} = (-1)^i)$

$\forall_{ai}(a - \text{целое} \mathbin{\&} \neg(a - \text{even}) \rightarrow (-1)^{a+i} = (-1)^{i+1})$

Антецеденты обрабатываются проверочными операторами. Во втором случае a отлично от 1. Уровень срабатывания равен 1.

$\forall_{mnpq}\left(n = 2p + 1 \mathbin{\&} m - \text{целое} \mathbin{\&} \neg(m - \text{even}) \rightarrow (-1)^{n/2+m/2} = (-1)^{p+1}(-1)^{[m/2]}\right)$

n - целочисленная константа. Уровень срабатывания равен 0. Первый антецедент выделен указателем "программа", второй и третий обрабатываются проверочными операторами.

$$\forall_{pn}\left((1 + (-1)^p)^n = (1 + (-1)^p)2^{n-1}\right)$$

n - натуральная константа, отличная от 1. Уровень срабатывания равен 1.

$\forall_{amn}(n - \text{rational} \mathbin{\&} \neg(\text{знаменатель}(n) - \text{even}) \rightarrow (-a)^n(-1)^m = a^n(-1)^{m+n})$

Уровень срабатывания равен 1.

$\forall_n(n - \text{целое} \rightarrow (-1)^n = (1 \text{ при } n - \text{even}, \text{ иначе } -1))$

Выражение расположено под частичным пределом по n. Уровень срабатывания равен 1.

37. Перенесение степени минус единицы из знаменателя в числитель.

$$\forall_{abn}\left(n - \text{rational} \mathbin{\&} \neg(\text{знаменатель}(n) - \text{even}) \rightarrow \frac{a}{b(-1)^n} = \frac{a(-1)^n}{b}\right)$$

Уровень срабатывания равен 2.

38. Разбор случаев в задаче на преобразование, имеющей минус единицу в целочисленной степени. Прием применяется в задачах на преобразование, имеющих цель "таблица". Это - очень редко встречающийся тип задач, возникающих при отыскании экстремумов. Предполагается, что ответ должен иметь вид условного выражения. Теорема приема имеет вид:

$\forall_n(n - \text{целое} \rightarrow n - \text{even} \lor \neg(n - \text{even}))$

Указатель "контрольвывода(степень(минус(1)x14))" определяет инициализацию приема при усмотрении выражения $(-1)^n$. Происходит вывод посылки $n - \text{even} \lor \neg(n - \text{even})$, сопровождаемой комментарием "разборслучаев". Уровень срабатывания равен 4.

39. Сокращение дроби с взаимно обратными экспонентами.

$$\forall_{abcd}\left(\frac{(\exp(a-b)+1)c}{(\exp(b-a)+1)d} = \frac{\exp(a-b)c}{d}\right)$$

Указатель "нормзнака(x2 минус плюс)" позволяет идентифицировать сначала b как сумму какого-то числа слагаемых, а затем - идентифицировать $-b$ путем изменения знаков всех этих слагаемых. Уровень срабатывания равен 3.

40. Вынесение за скобку общего степенного множителя при упрощении ответа.

$$\forall_{abcdemnp}\left(m-n=p\to\frac{a^m b}{c}+\frac{a^n d}{e}=a^n\left(\frac{a^p b}{c}+\frac{d}{e}\right)\right)$$

Прием применяется в задачах на преобразование, имеющих цель "известны". Такие задачи возникают при упрощении ответа задачи на описание, имеющей цель "известно" (например, в геометрии). Антецедент выделен указателем "идентификатор". Он вычитает показатели степени и получает некоторое выражение p, которое должно быть натуральной константой. Указатель "модификатор" обеспечивает проверку отсутствия других слагаемых суммы. Уровень срабатывания равен 3.

41. Специальные группировки для степени.

   (a) Группировка в виде степени с дробным основанием. Прием позволяет перенести экспоненциальную зависимость из знаменателя в числитель:

   $\forall_{icdef}(i - \text{rational}\ \&\ \neg(\text{знаменатель}(i) - \text{even})\ \&\ d - \text{rational}\ \&\ \neg(\text{знаменатель}(d) - \text{even}) \to e/fc^{di} = e(1/c^d)^i/f)$

   Выражение расположено внутри описателя "отображение" по переменной i. Эта переменная не входит в c, d. Внутри условных выражений прием блокируется. Уровень срабатывания равен 1.

   (b) Группировка под общий показатель степени при интегрировании. Приводимые ниже приемы применяются для преобразования подынтегрального выражения:

   $\forall_{abcde}(0 \le a\ \&\ 0 \le b \to a^{c+d}b^{c+e} = a^d b^e (ab)^c)$

   Переменная интегрирования входит в c и не входит в a, b, d, e. Уровень срабатывания равен 1.

$$\forall_{abcdefgh}\left(\frac{da^g}{h(ec^g+fb^g)}=\frac{d(a/b)^g}{h(e(c/b)^g+f)}\right)$$

   g - переменная интегрирования. Уровень срабатывания равен 3.

$$\forall_{abcdefghi}\left(\frac{ha^{dg+f}}{ie^{bg+c}}=\frac{ha^f(a^d/e^b)^g}{ie^c}\right)$$

   Переменная интегрирования входит в g и не входит в a, b, c, d, e, f. Уровень срабатывания равен 1.

   (c) Группировка под общий показатель степени при суммировании. Преобразуются выражения, расположенные под конечной суммой. Отличие от предыдущего пункта заключается в том, что для преобразования подынтегрального выражения создавалась отдельная вспомогательная задача. Уровень срабатывания приводимых приемов равен 0.

   $\forall_{abcdepq}(c - \text{целое}\ \&\ \neg(a = 0)\ \&\ \neg(b = 0) \to a^{pc+d}b^{qc+e} = a^d b^e (a^p b^q)^c)$

   Переменная суммирования входит в c и не входит в a, b, d, e. Дополнительно проверяются следующие требования:

i. Если одно из оснований степени a, b есть -1, то другое имеет заголовок "минус".

ii. Если списки параметров выражений a^p, b^q непусты, причем переменные одного из них встречаются только внутри преобразуемого выражения, то переменные другого списка должны встречаться тоже только внутри данного выражения.

iii. Ни одно из выражений a, b не имеет своим обобщенным сомножителем (достижимым из корня через "минус", "умножение", "дробь", "модуль" и через основания степеней) факториал.

iv. Не решается задача на разложение в ряд Тейлора.

$$\forall_{abcdefghi}\left(\frac{ha^{dg+f}}{ie^{bg+c}} = \frac{ha^f(a^d/e^b)^g}{ie^c}\right)$$

Переменная суммирования входит в g и не входит в a, b, c, d, e, f. Выполнены третье и четвертое требования предыдущего приема.

$\forall_{abcdepq}(c$ – rational & d – rational & e – rational & p – rational & q – rational & ¬(знаменатель(c) – even) & ¬(знаменатель(d) – even) & ¬(знаменатель(e) – even) & ¬(знаменатель(p) – even) & ¬(знаменатель(q) – even) $\rightarrow a^{pc+d}b^{qc+e} = a^d b^e (a^p b^q)^c)$

Переменная суммирования входит в c и не входит в a, b, d, e, p, q. Выполнены все требования, приведенные в первом приеме. Выражения p, q не имеют заголовка "минус".

(d) Группировка под общий показатель степени при усмотрении монотонности. Прием применяется во вспомогательной задаче на преобразование выражения, исследуемого на монотонную зависимость от заданной переменной x:

$$\forall_{abcdefghi}\left(\frac{ha^{dg+f}}{ie^{bg+c}} = \frac{ha^f(a^d/e^b)^g}{ie^c}\right)$$

Переменная x входит в g и не входит в a, b, c, d, e, f; $a \neq -1$. Выражения a, e не имеют своим обобщенным множителем факториал. Уровень срабатывания равен 0.

(e) Вынесение минус единицы из основания степени при анализе сходимости ряда.

$\forall_{ac}(0 \leq a \rightarrow (-a)^c = (-1)^c a^c)$

Прием применяется при анализе сходимости ряда, общий член которого содержит преобразуемое выражение. Варьируемая переменная (номер члена ряда) входит в c. Выражение a отлично от 1.

(f) Выделение дробной степени дробно-линейной функции при интегрировании. Приемы применяются во вспомогательной задаче на преобразование подынтегрального выражения. Уровень срабатывания их равен 2.

$\forall_{abcdefghijkn}(f = ng + h$ & $e = ni + j$ & $h + j = n$ & $h \leq j$ & ¬$(n$ – even$) \rightarrow (ak+b)^{f/n}(dk+c)^{e/n} = (ak+b)^g(dk+c)^{i+1}(ak+b/dk+c)^{h/n})$

k - переменная интегрирования. e, f, n - натуральные константы. Происходит вынесение из-под дробной степени натуральных степеней линейных функций. Антецеденты выполняют арифметические действия и выделены указателем "программа". Либо оба слагаемых b, c невырождены, либо выражение не является корневым и не является знаменателем корневой дроби, числитель которой не содержит k. При четном n применяется следующая версия приема:

$\forall_{abcdefghijkn}(f = ng + h \ \& \ e = ni + j \ \& \ h + j = n \ \& \ h \leq j \ \& \ n - \text{even} \rightarrow ((ak + b)^f (dk + c)^e)^{l/n} = |(ak + b)^{gl} (dk + c)^{(i+1)l}|(ak + b/dk + c)^{hl/n})$

Предполагается, что в случае $n = 2$ показатели e, f различны.

(g) Группировка членов с квадратом варьируемой координаты в уравнении кривой.

$\forall_{abx}(ax^2 + bx^2 = (a + b)x^2)$

Выражение входит в условие задачи на преобразование, имеющей цель "класс". Это означает, что результатом преобразований должен быть терм без описателей. Имеется внешний квантор существования по переменной x, причем его связывающая приставка не пересекается с параметрами коэффициентов a, b. Обычно такой квантор навешивается на уравнение кривой для описания множества ее точек. Прием способствует преобразованию уравнения к стандартному виду, позволяющему получить бескванторное описание множества. Уровень срабатывания равен 4.

(h) Группировка членов с радикалом в уравнении кривой.

$\forall_{abx}(a\sqrt{x} + b\sqrt{x} = (a + b)\sqrt{x})$

Аналогично предыдущему.

42. Использование посылки, дающей явное выражение для модуля основания степени.

$\forall_{ab}(|a| = b \rightarrow a^2 = b^2)$

Прием применяется в задачах на исследование, имеющих цель "известно". Антецедент берется в контексте замены. Выражение b не содержит числовых атомов, отличных от переменных и констант. Указатель "коммутативно(фикс(1))" определяет идентификацию посылки без перестановки частей равенства. Указатель "вид(фикс(0 1))" блокирует идентификацию квадрата a^2 в обобщенном смысле (как степени с четным показателем). Созданы две версии приема. Одна из них имеет своей точкой привязки выражение a^2, другая - инициирует работу при усмотрении посылки $|a| = b$. Последнее обеспечивается указателем "теквхожд(1)". Создание последней версии объясняется тем, что иногда посылка появляется значительно позже выражения a^2, и при ее отсутствии решатель будет слишком медлить с применением данного преобразования. Уровни срабатывания приемов равны 3.

43. Умножение на степень минус единицы.

$\forall_{mnk}(n - \text{целое} \rightarrow (-1)^{m+n} a^{n+k} = (-1)^{m+k} (-a)^{n+k})$

m, k - целочисленные константы. a - константное выражение. Уровень срабатывания равен 3.

$\forall_{abc}(\neg(\text{знаменатель}(c) - \text{even}) \ \& \ c - \text{rational} \rightarrow (b - a)^c (-1)^c = (a - b)^c)$

Уровень срабатывания равен 0.

Общая стандартизация утверждений

Для общей стандартизации утверждений, содержащих степенные выражения, используются следующие приемы:

1. Равенство степени нулю.

   $\forall_{ab}(a^b = 0 \leftrightarrow a = 0)$

   Указатель "сопровождение" разрешает выполнять преобразование даже в тех случаях, когда оно относится к сопровождающему по о.д.з. утверждению. Уровень срабатывания равен 0.

2. Неравенство со степенью в одной части и нулем в другой.

   $\forall_{ab}(\text{числитель}(b) - \text{even}\ \&\ b - \text{rational} \to a = 0 \leftrightarrow a^b \leq 0)$

   $\forall_{ab}(0 \leq a \to a = 0 \leftrightarrow a^b \leq 0)$

   Преобразования выполняются справа налево. Уровень срабатывания равен 0.

   $\forall_{ab}(0 \leq a \to 0 < a^b \leftrightarrow 0 < a)$

   $\forall_{ab}(\text{числитель}(b) - \text{even}\ \&\ b - \text{rational} \to 0 < a^b \leftrightarrow \neg(a = 0))$

   Здесь преобразования выполняются слева направо. Уровень срабатывания равен 0.

   $\forall_{ab}(b - \text{rational}\ \&\ \neg(\text{числитель}(b) - \text{even})\ \&\ \neg(\text{знаменатель}(b) - \text{even}) \to 0 \leq a^b \leftrightarrow 0 \leq a)$

   $\forall_{ab}(b - \text{rational}\ \&\ \neg(\text{числитель}(b) - \text{even})\ \&\ \neg(\text{знаменатель}(b) - \text{even}) \to 0 < a^b \leftrightarrow 0 < a)$

   Уровень срабатывания равен 1.

   $\forall ab(b - \text{rational}\ \&\ \neg(\text{знаменатель}(b) - \text{even}) \to 0 \leq a^b \leftrightarrow \neg(\text{числитель}(b) - \text{even})\ \&\ 0 \leq a\ \vee\ \text{числитель}(b) - \text{even})$

   $\forall_{ab}(b - \text{rational}\ \&\ \neg(\text{знаменатель}(b) - \text{even}) \to 0 < a^b \leftrightarrow \neg(a = 0)\ \&\ \text{числитель}(b) - \text{even}\ \vee\ \neg(\text{числитель}(b) - \text{even})\ \&\ 0 < a)$

   $\forall_{ab}(b - \text{rational}\ \&\ \neg(\text{знаменатель}(b) - \text{even}) \to a^b < 0 \leftrightarrow \neg(\text{числитель}(b) - \text{even})\ \&\ a < 0)$

   $\forall_{ab}(b - \text{rational}\ \&\ \neg(\text{знаменатель}(b) - \text{even}) \to a^b \leq 0 \leftrightarrow \neg(\text{числитель}(b) - \text{even})\ \&\ a \leq 0\ \vee\ a = 0\ \&\ \text{числитель}(b) - \text{even})$

   Заменяющие утверждения обрабатываются нормализаторами, во многих случаях исключающими одну из альтернатив дизъюнкций. Уровень срабатывания приемов равен 2.

   Усмотрение истинности неравенств выполняется следующими двумя приемами, срабатывающими на уровне 0 (неравенство заменяется на константу "истина"):

   $\forall_a(0 < a \to 0 < a^b)$

   $\forall_a(0 \leq a \to 0 \leq a^b)$

   Эти приемы представляют собой ускорители, введенные для специальных случаев - решается задача на описание, причем показатель степени есть простая дробь с четным знаменателем. Антецеденты обращаются к проверочным операторам. Для общего случая имеются отдельные приемы, приведенные в разделе, посвященном неравенствам.

3. Равенство степеней с одинаковыми показателями.

   $\forall_{abc}(\neg(\text{числитель}(b)-\text{even})\ \&\ b-\text{rational}\ \&\ \neg(\text{знаменатель}(b)-\text{even}) \to -c^b+a^b = 0 \leftrightarrow a-c=0)$

   $\forall_{abc}(\neg(b=0)\ \&\ 0 \le a\ \&\ 0 \le c \to -c^b + a^b = 0 \leftrightarrow a-c=0)$

   Эти приемы ориентированы на случаи, когда в результате общей стандартизации все ненулевые слагаемые оказались сгруппированы в одной части равенства. Уровень срабатывания равен 3. Добавлен еще один прием, предназначенный для упрощения теорем при логическом выводе:

   $\forall_{abc}(\neg(b=0)\ \&\ 0 \le a\ \&\ 0 \le c \to a^b = c^b \leftrightarrow a = c)$

   Уровень срабатывания здесь равен 1.

4. Строгое неравенство для разности степеней, имеющих одинаковое основание.

   $\forall_{abc}(0 < a\ \&\ 0 < 1-a \to 0 < -a^b + a^c \leftrightarrow 0 < b-c)$

   $\forall_{abc}(0 < a-1 \to 0 < -a^b + a^c \leftrightarrow 0 < c-b)$

   Уровень срабатывания равен 3.

5. Строгое неравенство для разности степеней, имеющих одинаковый показатель.

   $\forall_{abc}(0 < a\ \&\ 0 < b\ \&\ c < 0 \to 0 < -a^c + b^c \leftrightarrow 0 < a-b)$

   $\forall_{abc}(0 \le a\ \&\ 0 \le b\ \&\ 0 < c \to 0 < -a^c + b^c \leftrightarrow 0 < a-b)$

   $\forall_{abc}(c-\text{rational}\ \&\ c < 0\ \&\ \neg(\text{знаменатель}(c)-\text{even})\ \&\ \neg(\text{числитель}(c)-\text{even})\ \&\ a < 0\ \&\ b < 0 \to -a^c + b^c \leftrightarrow 0 < a-b)$

   $\forall_{abc}(c - \text{rational}\ \&\ 0 < c\ \&\ \neg(\text{знаменатель}(c) - \text{even})\neg(\text{числитель}(c) - \text{even}) \to 0 < -a^c + b^c \leftrightarrow 0 < b-a)$

   Уровень срабатывания равен 3.

6. Нестрогое неравенство для разности степеней, имеющих одинаковое основание.

   $\forall_{abc}(0 < a\ \&\ a-1 < 0 \to b-c \le 0 \leftrightarrow -a^b + a^c \le 0)$

   $\forall_{abc}(0 < a-1 \to c-b \le 0 \leftrightarrow -a^b + a^c \le 0)$

   Преобразования выполняются справа налево. Уровень срабатывания равен 3.

7. Нестрогое неравенство для разности степеней, имеющих одинаковый показатель.

   $\forall_{abc}(0 < a\ \&\ 0 < b\ \&\ c < 0 \to a-b \le 0 \leftrightarrow -a^c + b^c \le 0)$

   $\forall_{abc}(0 \le a\ \&\ 0 \le b\ \&\ 0 < c \to b-a \le 0 \leftrightarrow -a^c + b^c \le 0)$

   $\forall_{abc}(c-\text{rational}\ \&\ c < 0\ \&\ \neg(\text{знаменатель}(c)-\text{even})\ \&\ \neg(\text{числитель}(c)-\text{even})\ \&\ a < 0\ \&\ b < 0 \to a-b \le 0 \leftrightarrow -a^c + b^c \le 0)$

   $\forall_{abc}(c - \text{rational}\ \&\ 0 < c\ \&\ \neg(\text{знаменатель}(c) - \text{even})\neg(\text{числитель}(c) - \text{even}) \to b-a \le 0 \leftrightarrow -a^c + b^c \le 0)$

   Преобразования применяются справа налево. Уровень срабатывания равен 3.

8. Строгое неравенство для двух степеней с одинаковыми основаниями.

   $\forall_{abc}(0 < a\ \&\ a-1 < 0 \to a^b < a^c \leftrightarrow c < b)$

   $\forall_{abc}(0 < a-1 \to a^b < a^c \leftrightarrow b < c)$

   Уровни срабатывания равны 1 и 3.

9. Нестрогое неравенство для двух степеней с одинаковыми основаниями.

   $\forall_{abc}(0 < a \mathbin{\&} a - 1 < 0 \to b \leq c \leftrightarrow a^c \leq a^b)$

   $\forall_{abc}(0 < a - 1 \to c \leq b \leftrightarrow a^c \leq a^b)$

   Преобразования выполняются справа налево. Уровни срабатывания равны 1 и 3.

10. Строгое неравенство для двух степеней с одинаковыми показателями.

    $\forall_{abc}(c - \mathrm{rational} \mathbin{\&} 0 < c \mathbin{\&} \neg(\text{числитель}(c) - \mathrm{even}) \to 0 < b^c - a^c \leftrightarrow 0 < b - a)$

    Указатель "общаястепень(общаястепень x3 степень(x2 x3)степень(x1 x3))" определяет использование процедуры "общаястепень" для представления двух выражений в виде степеней с одинаковым показателем c. Этот показатель находится как общий множитель показателей степени сомножителей данных выражений. Уровень срабатывания равен 1.

11. Нестрогое неравенство для двух степеней с одинаковыми показателями.

    $\forall_{abc}(c - \mathrm{rational} \mathbin{\&} 0 < c \mathbin{\&} \neg(\text{числитель}(c) - \mathrm{even}) \to 0 \leq b^c - a^c \leftrightarrow 0 \leq b - a)$

    Аналогично предыдущему.

12. Строгое неравенство с единицей в одной части и степенью в другой.

    $\forall_{ab}(0 < b - 1 \to a < 0 \leftrightarrow b^a < 1)$

    $\forall_{ab}(0 < b - 1 \to 0 < a \leftrightarrow 1 < b^a)$

    $\forall_{ab}(0 < b \mathbin{\&} 0 < 1 - b \to 0 < a \leftrightarrow b^a < 1)$

    $\forall_{ab}(0 < b \mathbin{\&} 0 < 1 - b \to a < 0 \leftrightarrow 1 < b^a)$

    $\forall_{ab}(0 < a - 1 \to c < 0 \leftrightarrow 0 < -a^c + 1)$

    Преобразования выполняются справа налево. Уровни срабатывания равны 1 и 3 (кроме последнего приема, имеющего единственный уровень срабатывания 3).

13. Нестрогое неравенство с единицей в одной части и степенью в другой.

    $\forall_{ab}(0 < b \mathbin{\&} 0 < 1 - b \to 0 \leq a \leftrightarrow b^a \leq 1)$

    $\forall_{ab}(0 < b \mathbin{\&} 0 < 1 - b \to a \leq 0 \leftrightarrow 1 \leq b^a)$

    $\forall_{ab}(1 - b < 0 \to a \leq 0 \leftrightarrow b^a \leq 1)$

    $\forall_{ab}(1 - b < 0 \to 0 \leq a \leftrightarrow 1 \leq b^a)$

    Преобразования применяются справа налево; уровни срабатывания равны 1 и 3.

14. Получение явного выражения для параметра из простейшего степенного уравнения. Прием используется в задачах на исследование, имеющих цель "невыполнимо". Такие задачи возникают при попытках быстрой проверки несовместности нескольких новых утверждений со списком старых. Степенное уравнение разрешается относительно переменной x, чтобы после подстановки найденного значения в остальные посылки можно было усмотреть противоречие:

    $\forall_{abcx}(0 \leq x \mathbin{\&} \neg(a = 0) \mathbin{\&} \neg(b = 0) \to ax^b + c = 0 \leftrightarrow x = (-c/a)^{1/b} \mathbin{\&} ac \leq 0)$

    Переменная x не встречается в выражениях a, b, c. Либо a, либо b не равны 1. Уровень срабатывания равен 2.

15. Равенство степени единице.

    $\forall_{ab}(0 < a \ \& \ \neg(b = 0) \to a^b = 1 \leftrightarrow a = 1)$

    $\forall_n(n - \text{целое} \to (-1)^n = 1 \leftrightarrow n - \text{even})$

    Приемы срабатывают на уровне 1.

    $\forall_{cd}(\neg(c - 1 = 0) \ \& \ 0 < c \to d = 0 \leftrightarrow c^d = 1)$

    Преобразование применяется справа налево. Выражение d неконстантное. Уровень срабатывания равен 1.

    $\forall_{cd}(\neg(c + 1 = 0) \ \& \ c < 0 \ \& \ \text{числитель}(d) - \text{even} \ \& \ d - \text{rational} \to d = 0 \leftrightarrow c^d = 1)$

    Преобразование применяется справа налево. Выражение d неконстантное. Уровень срабатывания равен 2.

    $\forall_{cd}(\neg(c - 1 = 0) \ \& \ 0 < c \to -c^d + 1 = 0 \leftrightarrow d = 0)$

    $\forall_{cd}(\neg(c - 1 = 0) \ \& \ 0 < c \to -1 + c^d = 0 \leftrightarrow d = 0)$

    Уровень срабатывания приемов равен 3.

16. Возведение в квадрат равенств с радикалами, описывающих геометрическое место точек. Для возведения в квадрат уравнений с радикалами существует специальная серия приемов, которая будет рассмотрена в подразделе, посвященном уравнениям со степенными выражениями. Здесь мы приведем аналогичные приемы, относящиеся к уравнениям "без неизвестных", задающим в некоторой системе координат вид кривой. Переменные таких уравнений связаны внешним квантором существования. Исключение радикалов обеспечивает возможность последующего перехода к бескоординатному описанию вида кривой. Нижеследующие приемы применяются в задачах на преобразование, имеющих цель "класс" (исключение описателей и кванторов). Уровень срабатывания их равен 5.

    (a) Один радикал.

        $\forall_{abcd}(0 \le d \ \& \ 0 \le a \to a\sqrt{d} + b = c \leftrightarrow a^2d - b^2 + 2bc - c^2 = 0 \ \& \ 0 \le c - b)$

        $\forall_{abcd}(0 \le d \ \& \ a \le 0 \to a\sqrt{d} + b = c \leftrightarrow a^2d - b^2 + 2bc - c^2 = 0 \ \& \ c - b \le 0)$

        Равенство расположено под квантором существования, переменные которого пересекаются с параметрами выражения d. Среди слагаемых b нет другого радикала, пересекающегося с этими переменными.

    (b) Два радикала.

        $\forall_{abcdefq}(q = a^2e - c^2 + 2cd - b^2f + 2b(d - c)\sqrt{f} \ \& \ 0 \le a \to a\sqrt{e} + b\sqrt{f} + c = d \leftrightarrow q = d^2 \ \& \ 0 \le d - c - b\sqrt{f})$

        $\forall_{abcd}(a\sqrt{c} + b\sqrt{d} = 0 \leftrightarrow a^2c - b^2d = 0 \ \& \ ab \le 0)$

        Второй прием срабатывает, если не очевидна неотрицательность a. Иначе он поглощается первым приемом.

    (c) Четыре радикала.

        $\forall_{abcdefgh}(0 \le a \ \& \ 0 \le b \ \& \ c \le 0 \ \& \ d \le 0 \to a\sqrt{e} + b\sqrt{f} + c\sqrt{g} + d\sqrt{h} = 0 \leftrightarrow 2cd\sqrt{g}\sqrt{h} - 2ab\sqrt{e}\sqrt{f} + c^2g + d^2h - a^2e - b^2f = 0)$

17. Упрощение условия равенства суммы с радикалом нулю.

    $\forall_{ab}(a - \sqrt{b + a^2} = 0 \leftrightarrow b = 0 \ \& \ 0 \le a)$

$\forall_{ab}(a+\sqrt{b+a^2}=0 \leftrightarrow b=0 \ \& \ a\leq 0)$

Уровень срабатывания равен 4.

$\forall_{ab}(\sqrt{a-b}\sqrt{a+b}+b^2-a^2=0 \leftrightarrow a^2-b^2=0 \ \vee \ a^2-b^2=1)$

Уровень срабатывания равен 1.

$\forall_{abcdxp}(p=a^2-b^2c^2 \ \& \ \neg(p=0) \to ax+b\sqrt{c(x^2+d)}=0 \leftrightarrow$
$x=-b\mathrm{sg}(a)\sqrt{cd/p} \ \& \ 0\leq cdp)$

a,b,c,d - константные выражения. Уровень срабатывания равен 0.

18. Равенство двух радикалов.

    $\forall_{ab}(\sqrt{a}=\sqrt{b} \leftrightarrow a-b=0)$

    Уровень срабатывания равен 3.

    $\forall_{abc}(b-c=0 \to a\sqrt{b}=a\sqrt{c} \leftrightarrow \text{истина})$

    Прием применяется в задачах на исследование, имеющих цель "известно". Антецедент использует нормализатор раскрывания скобок "стандплюс". Уровень срабатывания равен 2.

19. Равенство степеней с одинаковыми основаниями.

    $\forall_{abc}(0<a \ \& \ \neg(a-1=0) \to a^b=a^c \leftrightarrow b=c)$

    Уровень срабатывания равен 0.

    $\forall_{abc}(\neg(a-1=0) \ \& \ 0<a \to -a^b+a^c=0 \leftrightarrow b-c=0)$

    Уровень срабатывания равен 3.

    $\forall_{abc}(\neg(a-1=0) \ \& \ \neg(a+1=0) \ \& \ b-\mathrm{rational} \ \& \ c-\mathrm{rational} \ \& \ \neg(\text{знаменатель}(b)-\mathrm{even}) \ \& \ \neg(\text{знаменатель}(c)-\mathrm{even}) \to a^b=a^c \leftrightarrow b=c)$

    Уровень срабатывания равен 1.

    $\forall_{amn}(a-\text{число} \ \& \ m-\text{целое} \ \& \ n-\text{целое} \to a^m=a^n \leftrightarrow m=n \ \vee \ a=0 \ \vee \ a=1 \ \vee \ a=-1 \ \vee \ m+n-\mathrm{even})$

    Уровень срабатывания равен 4.

20. Условие ненулевого значения суммы квадратов.

    $\forall_{ab}(\neg(a^2+b^2=0) \to \neg(a=0 \ \& \ b=0))$

    Уровень срабатывания равен 0.

21. Специальные равенства с радикалами.

    $$\forall_{abc}\left(\frac{a}{\sqrt{a^2+b^2}}=c \leftrightarrow c|b|=\sqrt{1-c^2}|a| \ \& \ 0\leq 1-c^2\right)$$

    Прием применяется в задачах на исследование. Выражение c не содержит неизвестных и отлично от переменной. a,b - переменные. Уровень срабатывания равен 2.

    $\forall_{abc}(0\leq a+b \ \& \ c\leq 0 \to \sqrt{a+b}+c=0 \leftrightarrow a-c^2+b=0)$

    b,c - десятичные константы. Равенство расположено в условии задачи на описание, не используемом для сопровождения по о.д.з. Уровень срабатывания равен 1.

$\forall_{abc}(0 \leq c \rightarrow \sqrt{a+b} = c \leftrightarrow a = c^2 - b)$

b, c - десятичные константы. Уровень срабатывания равен 1.

22. Исключение степени в параметрическом описании. Если в некотором параметрическом описании положительный параметр встречается только в виде степени с заданным показателем, то вместо него вводится положительный параметр, равный данной степени:

    $\forall_{ABn}(\exists_{xy}(A(x^n)\ \&\ x - \text{число}\ \&\ 0 < x\ \&\ B(y)) \leftrightarrow \exists_{xy}(A(x)\ \&\ x - \text{число}\ \&\ 0 < x\ \&\ B(y)))$

    Указатель "отображение(x26 x27)" задает функциональные переменные $A(\ldots)$, $B(\ldots)$. Идентификация начинается с усмотрения под квантором существования неравенства $0 < x$. Далее находится утверждение "$x -$ число" и идентифицируется конъюнкция $B(y)$ всех членов, не содержащих x. Остальные члены дают $A(x^n)$. Указатель "кортежпеременных(x24)" означает, что y идентифицируется с остатком (возможно, пустым) связывающей приставки квантора существования. Согласно указателю "контекст(позиция(x1 фикс(0 1))вид(x1 степень(x23 x14)))", под квантором существования находится выражение x^n и таким образом определяется n. Указатель "новаргумент(x26 набор(x23)фикс)" обеспечивает проверку того, что x встречается внутри $A(x^n)$ только в виде x^n. Указатель "содержится(x24 x26)" отменяет проверку невхождения переменных y в $A(x^n)$. Наконец, указатель "внешнийквантор(фикс(0 1))" отменяет попытку идентификации путем преобразования квантора общности к виду отрицания квантора существования. Прием срабатывает на уровне 3.

23. Исключение квантора существования для степеней минус единицы.

    $\forall_A(\exists_n(n - \text{целое}\ \&\ A((-1)^n)) \leftrightarrow A(1)\ \vee\ A(-1))$

    Указатель "отображение(x26)" вводит функциональную переменную A, указатель "новаргумент(x26 x14 фикс)" проверяет, что переменная n встречается внутри $A(\ldots)$ только в виде $(-1)^n$. Уровень срабатывания равен 1.

Усмотрение неотрицательности степени

$\forall_{ab}(0 \leq a \rightarrow 0 \leq a^b)$

$\forall_{ab}(\text{числитель}(b) - \text{even}\ \&\ b - \text{rational} \rightarrow 0 \leq a^b)$

Приемы заменяют неравенство на константу "истина". Уровень срабатывания равен 1.

Попытка разложения на множители основания степени

Обычно общая стандартизация степенных выражений состоит в том, чтобы разложить основание степени на множители, а затем представить степень произведения как произведение степеней. Первая часть этой последовательности действий реализуется следующими приемами:

$\forall_{abcd}(c = a + b\ \&\ d - \text{rational}\ \&\ \neg(\text{знаменатель}(d) - \text{even}) \rightarrow (a+b)^d = c^d)$

$\forall_{abcd}(c = a + b\ \&\ 0 \leq a + b \rightarrow (a+b)^d = c^d)$

Степенное выражение входит в условие задачи на преобразование. Первый антецедент обращается к нормализатору "видумножение". Проверяется, что либо c имеет мультипликативный заголовок ("умножение", "дробь", "степень", быть может, со знаком "минус"), либо исходная сумма имеет дробное слагаемое, а c не имеет. Задача не должна иметь цели либо комментария "длина", указывающих, что предпринимается завершающее упрощение выражения. Преобразуемая степень не должна находиться под описателем "отображение". Должен отсутствовать комментарий (Слагаемое $a+b$), указывающий, что данная сумма ранее возникла в результате раскрывания скобок. Первый прием проверяет также отсутствие комментария (Стандплюс $a+b$ p), где p - основание степени сомножителя выражения c. Второй прием дополнительно проверяет выполнение следующих требований:

1. Показатель степени не является простой дробью с нечетным знаменателем.
2. Если рассматриваемая степень имеет корневое вхождение и не решается задача на преобразование подынтегрального выражения, то c, с точностью до константного множителя, само должно являться степенным выражением.
3. Если решается задача на упрощение известных подтермов ответа задачи на описание и выражение c не имеет заголовка "степень", то рассматриваемое степенное выражение имеет некорневое вхождение и не расположено непосредственно под минусом.

Уровень срабатывания обоих приемов равен 4. Имеются также версии этих приемов, срабатывающие на уровне 5. В них, как и выше, рассматривается условие задачи на преобразование. Либо c имеет мультипликативный заголовок, либо исходная сумма имеет дробное слагаемое, а c не имеет. Кроме того, либо числитель выражения c должен иметь сомножитель вида степени, либо преобразуемая степень должна встречается в контексте $k(a+b)^d+m$, где некоторое слагаемое остаточной суммы m имеет общий множитель с c. Должны отсутствовать комментарии "длина" и (Слагаемое $a+b$). Преобразуемая степень не должна располагаться под описателем "отображение". Если преобразуемое вхождение корневое и c не имеет заголовка "степень", то второй прием блокируется.

Далее, предусмотрены версии рассматриваемых приемов, срабатывающие на уровнях 3 и 4 в условиях задач на описание. Если число неизвестных более одной, то приемы срабатывают на уровне 3, иначе - на уровне 4. Основание степени $a+b$ должно содержать неизвестные. Если показатель степени d равен 1/2, то c не должно иметь вида дроби.

Разложение на множители сомножителя основания степени

Если основание степени имеет вид произведения, но знаки сомножителей в контексте не очевидны и степень не удается представить в виде произведения степеней, то может оказаться полезным доразложение на множители сомножителей основания. Иногда это позволяет разбить сомножители на подгруппы определенного знака. Доразложение выполняется следующими приемами:

$\forall_{abcdef}(f = a+b \;\&\; c - \text{rational} \;\&\; \neg(\text{знаменатель}(c) - \text{even}) \;\&\; 0 \leq (a+b)^c d \rightarrow ((a+b)^c d)^e = (f^c d)^e)$

$\forall_{abcdef}(f = a+b \;\&\; 0 \leq a+b \;\&\; 0 \leq (a+b)^c d \rightarrow ((a+b)^c d)^e = (f^c d)^e)$

Приемы применяются к условию задачи на преобразование. Первый антецедент обращается к нормализатору "видумножение". Результат f должен иметь мультипликативный заголовок. Задача не имеет комментария "длина". Уровень срабатывания равен 1. Созданы также версии этих приемов, срабатывающие на уровне 5.

Усмотрение неотрицательности слагаемого в сумме с радикалом

При преобразовании двойных радикалов вида $\sqrt{a+b\sqrt{c}}$ может оказаться полезным усмотрение неотрицательности выражения a:

$\forall_{abcd}(d^2 = a^2 - b^2c \ \& \ 0 \leq a + b\sqrt{c} \rightarrow 0 \leq a)$

Прием выполняет вывод в посылках задачи на преобразование. Он инициируется с помощью указателя "контрольвывода(степень(плюс(x1 умножение(x2 степень(x3 дробь(1 2)))) дробь(x5 умножение(2 x6))))", усматривающего двойной радикал в условии задачи. Указатель "внимание(теквхожд)" понижает до нуля веса всех термов задачи, содержащих этот радикал. Антецедент выделен указателем "идентификатор". Правая часть равенства обрабатывается сначала нормализатором раскрывания скобок "стандплюс", затем нормализатором "видумножение". Проверяется, что результат является полным квадратом.

Усмотрение полного квадрата в выражении с радикалом

Иногда удается избавиться от двойных радикалов вида $\sqrt{c+a\sqrt{b}}$, усмотрев в выражении $c+a\sqrt{b}$ полный квадрат. Для этого создан прием, выполняющий следующие действия. Сначала предпринимается доразложение на множители выражения $\sqrt{b}$, и среди них выбирается некоторый радикал $\sqrt{h}$. Далее подбираются такие выражения x, y, что $(x\sqrt{h}+y)^2 = c+a\sqrt{b}$, причем c есть сумма квадратов, а $a\sqrt{b}$ - удвоеннное произведение. Решив несложную систему уравнений относительно x, y, приходим к следующей теореме приема:

$\forall_{abcdefgh}(d^2 = c^2 - a^2g^2h \ \& \ (2(c-d) = he^2 \vee 2(c+d) = he^2) \ \& \ fe = 2ag \ \& \ \sqrt{b} = g\sqrt{h} \rightarrow c + a\sqrt{b} = (e\sqrt{h}/2 + f/2)^2)$

Антецеденты выделены указателями "идентификатор". Сначала четвертый антецедент обращается к оператору "видумножение" для разложения на множители b и к нормализатору "нормстепень" для последующей стандартизации степени. В полученном произведении выбирается некоторый сомножитель вида $\sqrt{h}$, а остальные сомножители определяют выражение g. Далее первый антецедент упрощает выражение $c^2 - a^2g^2h$, используя нормализаторы "стандплюс"и "видумножение". Проверяется, что результат есть полный квадрат некоторого выражения d. Второй антецедент рассматривает два подслучая. В каждом из них левая часть равенства обрабатывается нормализатором "видумножение"; проверяется, что результат делится на h, а частное представляет собой полный квадрат некоторого выражения e. Наконец, третий антецедент проверяет делимость выражения $2ag$ на e и идентифицирует выражение f. Прием применяется к условию задачи на преобразование либо описание, причем в последнем случае преобразуемое выражение должно содержать неизвестные, а число неизвестных задачи должно быть равно 1. Проверяется, что выражение $c + a\sqrt{b}$ является основанием степени, имеющей своим показателем дробь с делящимся на 2 знаменателем. Проверяется также, что c не имеет слагаемых с радикалами. Уровень срабатывания равен 1.

В случае, когда перед выражением $c+a\sqrt{b}$ находится минус, применяется другая версия приема:

$\forall_{abcdefgh}(d^2 = c^2 - a^2g^2h \;\&\; (2(-c-d) = he^2 \;\vee\; 2(-c+d) = he^2) \;\&\; fe = -2ag \;\&\; \sqrt{b} = g\sqrt{h} \rightarrow -(c + a\sqrt{b}) = (e\sqrt{h}/2 + f/2)^2)$

Иногда удается исключить двойные радикалы, домножив числитель и знаменатель дроби на сумму, сопряженную со знаменателем:

$\forall_{abc}(a^2 - b^2 = c^2 \;\&\; 0 \leq a + b \;\&\; 0 \leq c \rightarrow \sqrt{(a+b)/(a-b)} = (a+b)/c)$

Здесь хотя бы одно из выражений a, b должно иметь своим сомножителем квадратный радикал. Первый антецедент обрабатывает выражение $a^2 - b^2$ нормализатором "видумножение" и усматривает, что результат является полным квадратом. Прием срабатывает на уровне 3.

Попытка перемножения двух сумм с одинаковыми радикалами

Если перемножаются две суммы с одинаковми радикалами, то при раскрывании скобок выражение может существенно упроститься. Для реализации такого упрощения созданы несколько приемов.

$\forall_{abcdefg}(g = adb + cd\sqrt{b} + ae\sqrt{b} + ce \;\&\; f - \text{rational} \;\&\; \neg(\text{знаменатель}(f) - \text{even}) \rightarrow (a\sqrt{b} + c)^f(d\sqrt{b} + e)^f = g^f)$

Прием применяется к условию задачи на преобразование. Антецедент обрабатывает результат раскрывания скобок нормализаторами "стандплюс" и "видумножение" (последнее - только для неконстантных выражений), а также задачей на упрощение, решаемой до максимального уровня 4. Проверяется, что заменяющий терм короче заменяемого. Если при раскрывании скобок радикалы взаимно уничтожаются, причем g - не десятичная константа, то прием блокируется. Эта ситуация означает переход от произведения суммы выражений на их разность к разности квадратов, обеспечиваемый другими приемами. В зависимости от целевой установки задачи, уровень срабатывания равен 3 либо 4. Создана версия приема, применяемая в задачах на исследование. Здесь уровень срабатывания равен 1. Дополнительно требуется, чтобы преобразуемое выражение не содержало неизвестных и невырожденных числовых атомов, причем каждое выражение, встречающееся под степенью с дробным показателем в одной из сумм $a\sqrt{b} + c, d\sqrt{b} + e$, встречалось бы под степенью с дробным показателем в другой сумме.

$\forall_{abcdefgh}(h = a^2bd\sqrt{b} + 2abcd + c^2d\sqrt{b} + a^2be + 2ace\sqrt{b} + c^2e \rightarrow (a\sqrt{b}+c)^{2f/g}(d\sqrt{b}+e)^{f/g} = h^{f/g})$

Прием отличается от предыдущего тем, что при раскрывании скобок первая сумма с радикалом возводится в квадрат. Антецедент использует те же нормализаторы, что и выше. f, g - натуральные константы, причем g нечетно. Уровень срабатывания равен 5. Чтобы учесть случай, когда второй множитель возводится в степень 1, создана версия приема без дробных показателей:

$\forall_{abcdefh}(h = a^2bd\sqrt{b} + 2abcd + c^2d\sqrt{b} + a^2be + 2ace\sqrt{b} + c^2e \rightarrow (a\sqrt{b}+c)^{2f}(d\sqrt{b}+e)^f = h^f)$

Уровень срабатывания ее тоже равен 5. Для произведения суммы константных выражений с радикалами на их разность созданы следующие два приема, срабатывающие на уровне 0:

$\forall_{abcde}((a\sqrt{b} + c\sqrt{d})^e(a\sqrt{b} - c\sqrt{d})^e = (a^2b - c^2d)^e)$

$\forall_{abcd}((a+b\sqrt{c})^d(a-b\sqrt{c})^d=(a^2-b^2c)^d)$

Здесь все переменные в основании степеней идентифицированы с десятичными константами.

Разбор случаев по знаку сомножителя под радикалом при упрощении выражений

Чтобы усмотреть полный квадрат в выражении вида $p+q\sqrt{ab}$, расположенным под радикалом, может понадобиться разбор случаев по знакам сомножителей a, b. После уточнения этих знаков выражение $\sqrt{ab}$ преобразуется в произведение радикалов, и далее полный квадрат усматривается, как описано выше. Теорема приема, выполняющего разбор случаев, имеет следующий вид:

$\forall_{ab}(0\leq ab\to 0\leq a\ \&\ 0\leq b\ \vee\ a\leq 0\ \&\ b\leq 0)$

Указатель "контрольвывода(степень(умножение(x1 x2)x3))" инициирует применение приема при усмотрении выражения $(ab)^c$. Проверяется, что это выражение расположено под внешним радикалом четной степени. Проверяется также, что текущая задача имеет тип "преобразовать", выражение расположено в ее условии и не имеет места этап завершающего упрощения ответа. Прием заносит дизъюнкцию в посылки и сопровождает ее комментарием "разборслучаев", инициирующим немедленный разбор случаев. Уровень срабатывания равен 9.

$\forall_b(\neg(b=0)\to b<0\ \vee\ 0<b)$

Этот прием добавлен специально для задач на преобразование подынтегральных выражений. Он инициируется указателем "контрольвывода(дробь(x1 x2))" при усмотрении дробного выражения a/b, числитель которого содержит переменную интегрирования, а знаменатель - не содержит. Проверяется, что данная дробь расположена непосредственно под квадратным радикалом. Уровень срабатывания равен 5.

Решение простейшего степенного уравнения

$\forall_{abc}(\neg(\text{числитель}(b)-\text{even})\ \&\ b-\text{rational}\ \&\ \neg(\text{знаменатель}(b)-\text{even})\ \&\ c-\text{число}\to c=a^b\leftrightarrow a=c^{1/b})$

Прием применяется к условию задачи на описание либо посылке задачи на исследование. Вхождение уравнения - не обязательно корневое. Однако, заголовки надтермов уравнения допускаются лишь трех типов - "и", "или", "не". Выражение a содержит неизвестные, выражения b, c - не содержат. Указатель "общаястепень(общаястепень x2 степень(x1 x2))" позволяет усматривать степенное выражение a^b в произведении нескольких степеней. При этом b идентифицируется как наибольший общий делитель показателей степени сомножителей. Уровень срабатывания равен 1.

$\forall_{abc}(\neg(b=0)\ \&\ 0\leq a\ \&\ c-\text{число}\to a^b=c\leftrightarrow 0\leq c\ \&\ a=c^{1/b})$

Аналогично предыдущему. Уровень срабатывания равен 1.

$\forall_{abcd}(b-\text{rational}\ \&\ \text{числитель}(b)-\text{even}\ \&\ \neg(b=0)\ \&\ d=c^{1/b}\ \&\ c-\text{число}\to a^b=c\leftrightarrow 0\leq c\ \&\ (a=d\ \vee\ a=-d))$

Этот прием инициирует разбор случаев, и его применение сопровождается дополнительными ограничениями. Он не используется в задачах на исследование, имеющих цель "известно". Выражение a не должно представлять собой произведение, часть сомножителей которого известны. Задача не должна иметь линейного уравнения относительно своей единственной неизвестной. Некоторые ограничения накладываются также в случае решения дифференциальных уравнений. Как и в предыдущих приемах, используется указатель "общаястепень". Четвертый антецедент обращается к нормализаторам "нормстепень" и "сбросмодуля". Последний позволяет отбрасывать знак модуля у сомножителей, так как он не нужен при рассмотрении альтернатив d, $-d$. Уровень срабатывания равен 1. Для решения дифференциальных уравнений создана еще одна версия данного приема, срабатывающая на уровне 7.

Если показатель степени может обращаться в ноль, применяются следующие приемы:

$$\forall_{abc}(0 < a \;\&\; c - \text{число} \to c = a^b \leftrightarrow \neg(b = 0) \;\&\; a = c^{1/b} \;\&\; 0 < c \;\vee\; b = 0 \;\&\; c = 1)$$

$$\forall_{abc}(0 \leq a \;\&\; (\neg(a = 0) \;\vee\; 0 < b) \to a^b = c \leftrightarrow 0 < c \;\&\; \neg(b = 0) \;\&\; a = c^{1/b} \;\vee\; c = 0 \;\&\; a = 0 \;\&\; 0 < b \;\vee\; b = 0 \;\&\; c = 1 \;\&\; 0 < a)$$

Первый прием срабатывает на уровне 4, второй - на уровне 7. В заключение приведем два приема, срабатывающие в задачах на исследование, имеющих цель "известно" (первый из них - также в посылках задач на доказательство):

$$\forall_{abc}(\neg(b = 0) \;\&\; 0 \leq a \;\&\; c - \text{число} \to a^b = c \leftrightarrow a = c^{1/b} \;\&\; 0 \leq c)$$

$$\forall_{abc}(b - \text{rational} \;\&\; a \leq 0 \;\&\; \text{числитель}(b) - \text{even} \;\&\; c - \text{число} \to a^b = c \leftrightarrow 0 \leq c \;\&\; a = -c^{1/b})$$

Первый прием имеет уровни срабатывания 5 и 6, второй - уровень 3.

Решение квадратного уравнения

$$\forall_{abcde}(e = b^2 + 4ac \to ad^2 + bd = c \leftrightarrow \neg(a = 0) \;\&\; 0 \leq e \;\&\; (d = (\sqrt{e} - b)/2a \;\vee\; d = -(\sqrt{e} + b)/2a) \;\vee\; bd = c \;\&\; a = 0)$$

Прием применяется к равенству, представляющему собой условие задачи на описание. Выражения a, b, c не содержат неизвестных, d - содержит неизвестные. Предшествующие решению квадратного уравнения срабатывания приемов переносят все члены с неизвестными в левую часть, а известные члены в правую. Поэтому аналогичная группировка предпринята и в рассматриваемой теореме. Антецедент обрабатывает дискриминант нормализатором разложения на множители "видумножение". Этим же нормализатором обрабатываются коэффициент a и числители дробей под дизъюнкцией. Радикал $\sqrt{e}$ обрабатывается нормализаторами "нормстепень" и "сбросмодуля", так как при его упрощении часто возникают сомножители с избыточным заголовком "модуль". Если число неизвестных задачи более одной, то при наличии целочисленных неизвестных, входящих в уравнение, прием блокируется. Для преобразования дифференциальных уравнений прием применяется только в случае, когда дискриминант e оказывается полным квадратом. Если задача имеет другие уравнения, то заменяющее утверждение снабжается комментарием "разборслучаев", вызывающим немедленный разбор случаев по дизъюнкции для d. Уровень срабатывания равен 2 для задач с единственной неизвестной и 1 для задач с несколькими неизвестными. Создана версия приема, применяемая в задачах на исследование, имеющих цель "известно". Здесь требуется, чтобы неизвестное выражение d содержало расстояние между точками либо координату точки. Уровень срабатывания равен 3. Для специального случая, возникающего в геометрических задачах, создан еще один прием, срабатывающий на уровне 0:

$\forall_{abcdx}(d = -a^2+2c-2b \,\&\, 0 \leq d \rightarrow x^2+(a-x)^2+b = c \leftrightarrow x = (a-\sqrt{d})/2 \vee x = (a+\sqrt{d})/2)$

Усмотрение существования либо несуществования решения квадратного уравнения обеспечивается серией приемов, имеющих заголовок "связка". Они срабатывают в условиях задачи на описание. Рассмотрим более подробно первый прием серии:

$\forall_{abc}(\exists_d(d - \text{число} \,\&\, ad^2 + bd = c) \leftrightarrow 0 \leq b^2 + 4ac)$

Переменная d под квантором существования идентифицируется с несущественной неизвестной задачи. При этом все условия задачи, содержащие данную неизвестную, идентифицируются с подкванторными утверждениями. Дополнительно проверяется, что d не входит в a, b, c. Прием удаляет содержащие d условия задачи и заменяет их на требование неотрицательности дискриминанта. При редактировании параметрического описания прием блокируется. Уровень срабатывания равен 0. Перечислим остальные приемы серии:

$\forall_{abcd}(0 < a \,\&\, 2ad + b \leq 0 \rightarrow \exists_x(x - \text{число} \,\&\, ax^2 + bx = c \,\&\, d \leq x) \leftrightarrow 0 \leq b^2 + 4ac)$

$\forall_{abcmn}(m = 2n \,\&\, a < 0 \,\&\, 0 < c \rightarrow \exists_x(x - \text{число} \,\&\, ax^m + bx^n = c \,\&\, 0 < x) \leftrightarrow 0 < b \,\&\, 0 \leq b^2 + 4ac)$

$\forall_{abc}(0 < a \,\&\, c < 0 \rightarrow \exists_x(x - \text{число} \,\&\, ax^2 + bx = c \,\&\, 0 < x) \leftrightarrow 0 \leq b^2 + 4ac \,\&\, b < 0)$

Они усматривают существование решения квадратного либо биквадратного уравнения при наличии дополнительного неравенства. Уровни срабатывания равны 0. Наконец, создан прием, исключающий квантор существования над квадратным уравнением:

$\forall_{abc}(\exists_x(ax^2 + bx + c = 0 \,\&\, x - \text{число}) \leftrightarrow \neg(a = 0) \,\&\, 0 \leq b^2 - 4ac \;\vee\; a = 0 \,\&\, (c = 0 \;\vee\; \neg(b = 0)))$

Это обычный прием замены, имеющий заголовок "второйтерм". Уровень срабатывания равен 1.

Уравнения с радикалами

Для решения уравнений с радикалами созданы приемы возведения в квадрат или в куб. При большом числе слагаемых с радикалами радикалов возведение в степень становится неэффективным. Поэтому ограничиваемся случаями не более чем четырех слагаемых с квадратными либо трех слагаемых с кубическими радикалами, рассматривая их по отдельности. Для краткости ниже говорим не о числе слагаемых с радикалами, а о числе радикалов, подразумевая первое.

1. Возведение в квадрат уравнения с одним неизвестным радикалом.

   $\forall_{abcdp}(0 \leq d \rightarrow ad^{p/2} + b = c \leftrightarrow a^2d^p - b^2 + 2bc = c^2 \,\&\, 0 \leq (c - b)a)$

   Прием применяется к уравнению задачи на описание. p - натуральная константа (очевидно, нечетная после предшествующей стандартизации). Выражение d содержит неизвестные, c - не содержит. Выражение b не имеет слагаемых вида $kr^{m/2n}$, где r содержит неизвестные. Если уравнение имеет более одной неизвестной, b отлично от 0 и имеется другое уравнение с не менее чем двумя неизвестными радикалами, то прием блокируется. В этой ситуации предпочтительнее сначала возвести в квадрат второе уравнение. Прием блокируется также, если b содержит неизвестный логарифм, а d не содержит. Несколько фильтров ограничивают применение приема при решении дифференциальных уравнений. Указатель "удалениеусловия(условие(легковидеть(не(равно(x1

0))))меньшеилиравно(0 x4))" вызывает удаление условия $0 \le d$, если оно не используется более для сопровождения по о.д.з. Предварительно проверяется, что $a \ne 0$. После возведения в квадрат обе части уравнения обрабатываются нормализатором раскрывания скобок "стандплюс". Уровень срабатывания равен 5. Для преобразования дифференциальных уравнений с радикалом созданы еще два приема возведения в квадрат:

$$\forall_{abcd}(0 \le d \to a\sqrt{d} + b = c \leftrightarrow a^2d - b^2 + 2bc = c^2 \mathbin{\&} 0 \le (c-b)a)$$

$$\forall_{ab}(\sqrt{a} = b \leftrightarrow a = b^2)$$

Первый из них срабатывает на уровне 11 при преобразовании самого дифференциального уравнения, второй - на уровне 3 при редактировании его ответа. Наконец, в задачах на исследование, не имеющих цели "известно", применяется следующий прием:

$$\forall_{abcd}(a\sqrt{d} + b = c \to a^2d - b^2 + 2bc - c^2 = 0)$$

Выражения a, c известны. Выражение d линейно относительно некоторой неизвестной, не входящей в b, причем существует другое уравнение, линейное относительно данной неизвестной. Выражение b не имеет слагаемых вида $kr^{m/2n}$. Уровень срабатывания равен 5.

2. Возведение в квадрат уравнения с двумя неизвестными радикалами. Уравнение с двумя радикалами можно возводить в квадрат двумя способами: сгруппировать в одной части ровно два слагаемых с радикалами, перенеся остальные слагаемые в правую часть, либо сгруппировать в каждой части по одному радикалу. Каждый из них дает не более одного слагаемого с радикалом, но в определенных ситуациях один из способов оказывается существенно выгоднее другого. Начнем с приема, реализующего первый способ:

$$\forall_{abcdefg}(g = a^2e + b^2f \to a\sqrt{e} + b\sqrt{f} + c = d \leftrightarrow g + 2ab\sqrt{e}\sqrt{f} - c^2 + 2cd = d^2 \mathbin{\&} 0 \le (a\sqrt{e} + b\sqrt{f})(d-c))$$

Преобразуемое уравнение является условием задачи на описание. Выражения e, f различны и содержат неизвестные, выражение d не содержит неизвестных. Выражение c не имеет слагаемых вида $kr^{m/2n}$ при неизвестном r. Антецедент упрощает сумму возведенных в квадрат слагаемых с радикалами, используя нормализатор раскрывания скобок "стандплюс". Прием рассчитан на те случаи, когда в результате сокращений эта сумма теряет неизвестные. Другой возможностью его применения является существование уравнения, содержащего произведение $\sqrt{e}\sqrt{f}$. Уровень срабатывания равен 5.

$$\forall_{abcdefq}(q = a^2e - c^2 + 2cd - b^2f + 2b(d-c)\sqrt{f} \to a\sqrt{e} + b\sqrt{f} + c = d \leftrightarrow q = d^2 \mathbin{\&} 0 \le (d - c - b\sqrt{f})a)$$

Условия на e, f, d, c те же, что и выше. Выражение e не короче выражения f - таким образом после возведения в квадрат под радикалом оказывается более короткое выражение. Слагаемое с радикалом e остается в левой части; прочие слагаемые переносятся в правую часть и выполняется возведение в квадрат. Затем в левую часть возвращаются все слагаемые кроме d^2, и полученная сумма упрощается антецедентом. Здесь снова используется нормализатор "стандплюс". Если число неизвестных слагаемых в новой левой части уравнения оказалось больше 8, прием блокируется. Уровень срабатывания равен 5. Заметим, что здесь имеется один из немногих в решателе случаев, когда важен порядок компиляции приемов. Оба они срабатывают на одном и том же уровне, и если сначала произойдет обращение к программе второго приема, то сработает он, даже если ситуация более подходит для срабатывания первого приема. Правило определения очередности компиляции простое: если программы двух приемов имеют общую начальную часть, а затем разветвляются, то перекомпилированный последним прием будет выполняться после другого. Таким образом, для сохранения правильной очередности в нашем случае сначала должен компилироваться первый прием, а затем второй. В заключение приведем прием, выполняющий возведение в квадрат уравнения с двумя радикалами для задач на исследование, имеющих цель "известно":

$$\forall_{abcd}(0 < a \ \& \ 0 < c \to a\sqrt{b} - c\sqrt{d} = 0 \leftrightarrow a^2b - c^2d = 0)$$

Уравнение не должно иметь неконстантные числовые атомы, отличные от переменных. Выражения a, c - константные, причем общее число переменных в уравнении не более двух. Уровень срабатывания равен 4.

3. Возведение в квадрат уравнения с тремя неизвестными радикалами. В этом случае перед возведением в квадрат два радикала остаются в левой части, а остальные переносятся в правую часть:

   $$\forall_{abcdefgh}(a\sqrt{e} + b\sqrt{f} + c\sqrt{g} + d = h \leftrightarrow a^2e + b^2f + 2ab\sqrt{e}\sqrt{f} - c^2g + 2c(h-d)\sqrt{g} + 2dh - d^2 = h^2 \ \& \ 0 \leq (a\sqrt{e} + b\sqrt{f})(h - d - c\sqrt{g}))$$

   Уравнение представляет собой условие задачи на описание. Выражения e, f, g содержат неизвестные, выражение h известно. Выражение d не имеет слагаемых вида $kr^{m/2n}$ при неизвестном r. В правую часть переносится радикал $\sqrt{g}$; он выбирается как самый длинный из трех радикалов. Если задача имеет единственную неизвестную, то прием срабатывает на уровне 5, иначе - на уровне 6. В последнем случае на уровне 5 предпринимается попытка применить другую версию приема, в которой радикал $\sqrt{g}$ содержит неизвестную, не встречающуюся в двух других радикалах. Лишь при невозможности такого выбора радикала применяется первая версия приема.

4. Возведение в квадрат уравнения с четырьмя неизвестными радикалами.

   $$\forall_{abcdefghi}(i = a^2e + b^2f - c^2g - d^2h \to a\sqrt{e} + b\sqrt{f} + c\sqrt{g} + d\sqrt{h} = 0 \leftrightarrow 2cd\sqrt{g}\sqrt{h} - 2ab\sqrt{e}\sqrt{f} = i \ \& \ (a\sqrt{e} + b\sqrt{f})(c\sqrt{g} + d\sqrt{h}) \leq 0)$$

   В этом случае группировка радикалов выполняется так, чтобы после возведения в квадрат произошли сокращения, устраняющие неизвестные из всех слагаемых, кроме удвоенных произведений. Антцедент обрабатывает разность двух сумм квадратов с помощью нормализатора раскрывания скобок "стандплюс". Затем проверяется, что результат i не содержит неизвестных. Уровень срабатывания равен 5.

5. Возведение в куб уравнения с одним кубическим радикалом.

   $$\forall_{abcd}(a\sqrt[3]{b} + c = d \leftrightarrow a^3b + c^3 - 3c^2d + 3cd^2 = d^3)$$

Преобразуется условие задачи на описание. Выражение b содержит неизвестные, d - не содержит. Отсутствует слагаемое выражения c, имеющее своим сомножителем степень с дробным показателем и неизвестным основанием. Если преобразуется дифференциальное уравнение, то варьируемая переменная должна входить в него только как аргумент неизвестных функций либо их производных. Уровень срабатывания равен 5 для задач с единственной неизвестной и 6 для задач с несколькими неизвестными.

6. Возведение в куб уравнения с двумя кубическими радикалами.

   $\forall_{abcde}(a\sqrt[3]{b}+c\sqrt[3]{d}=e \leftrightarrow a^3b+c^3d+3ac\sqrt[3]{b}\sqrt[3]{d}e=e^3 \;\&\; (\neg(a^3b-c^3d=0) \vee \neg(c^3d+e^3=0)) \;\vee\; a\sqrt[3]{b}=0 \;\&\; c\sqrt[3]{d}=0 \;\&\; e=0)$

   Теорема приема требует некоторых пояснений. Если просто возвести обе части уравнения в куб, то получится $a^3b+c^3d+3ac^2\sqrt[3]{b}\sqrt[3]{d^2}+3a^2c\sqrt[3]{b^2}\sqrt[3]{d}=e^3$. Теперь можно сгруппировать два последних произведения, вынеся за скобки $3ac\sqrt[3]{b}\sqrt[3]{d}$, а оставшуюся в скобках сумму заменить на e. Таким образом, новое уравнение является следствием старого. Чтобы выяснить, при каких условиях старое уравнение вытекает из нового, вычтем обе части последнего из тождества для куба суммы $a\sqrt[3]{b}+c\sqrt[3]{d}$. Получим $(a\sqrt[3]{b}+c\sqrt[3]{d})^3-e^3=3ac\sqrt[3]{b}\sqrt[3]{d}(a\sqrt[3]{b}+c\sqrt[3]{d}-e)$. После разложения на множители левой части как разности кубов имеем общий множитель $(a\sqrt[3]{b}+c\sqrt[3]{d}-e$ обеих частей уравнения. Равенство этого множителя нулю эквивалентно истинности исходного уравнения. Сокращая на него и перенося все члены в левую часть, приходим к равенству вида $x^2+y^2+z^2-xy+yz+xz=0$, где x, y, z обозначают, соответственно, выражения $a\sqrt[3]{b}, c\sqrt[3]{d}$ и e. Нетрудно видеть, что левая часть есть сумма трех квадратов - полуразности x, y и полусумм x, z; y, z. Таким образом, если выполнено хотя бы одно из условий $\neg(a^3b-c^3d=0), \neg(c^3d+e^3=0)$, то данная сумма квадратов заведомо ненулевая, и старое уравнение является следствием нового. Однако, надо еще убедиться в том, что вся заменяющая дизъюнкция является следствием старого уравнения. Если оба приведенных выше отрицания равенств нарушены и выполнено старое уравнение, то $a\sqrt[3]{b}=c\sqrt[3]{d}=-e$, откуда вытекает $a\sqrt[3]{b}=c\sqrt[3]{d}=e=0$.

   Прием применяется к условию задачи на описание; выражения b, d содержат неизвестные, а e не содержит. Уровень срабатывания равен 5.

7. Возведение в куб уравнения с тремя кубическими радикалами.

   $\forall_{abcdef}(a\sqrt[3]{d}+b\sqrt[3]{e}+c\sqrt[3]{f}=0 \rightarrow a^3d+b^3e+c^3f-3abc\sqrt[3]{d}\sqrt[3]{e}\sqrt[3]{f}=0)$

   Прием принципиально отличается от предыдущего: вместо эквивалентного преобразования исходного уравнения происходит занесение в список условий задачи на описание нового уравнения. Это упрощает вид теоремы приема, но несколько усложняет решение, так как после нахождения значений неизвестных приходится делать проверку, подставляя их в исходное уравнение. В качестве упражнения можно рекомендовать создать версию данного приема, выполняющую эквивалентное преобразование. Выражения b, e, f содержат неизвестные. Уровень срабатывания равен 5.

8. Возведение в квадрат уравнения с одним радикалом четвертой степени.

   $\forall_{abcd}(a\sqrt[4]{b}+c=d \leftrightarrow a^2\sqrt{b}-c^2-d^2+2cd=0 \;\&\; 0\leq ab(d-c))$

Выражение b содержит неизвестные, а выражение d не содержит. Выражение c не имеет слагаемых, сомножители которых являлись бы степенями с дробным показателем отличным от $1/2$ и неизвестным основанием. Число неизвестных квадратных радикалов, являющихся сомножителями слагаемых выражения c, не более 3. Прием срабатывает на уровне 5.

9. Двойное возведение в квадрат уравнения с двумя радикалами четвертой степени.

   $\forall_{abcdef}(f = c^4a + d^4b \rightarrow c\sqrt[4]{a} + d\sqrt[4]{b} = e \leftrightarrow 2c^2d^2\sqrt{a}\sqrt{b} - 4cde^2\sqrt[4]{a}\sqrt[4]{b} = f - e^4 \ \& \ 0 \leq e(c\sqrt[4]{a} + d\sqrt[4]{b}) \ \& \ 0 \leq e^2 - 2cd\sqrt[4]{a}\sqrt[4]{b})$

   Выражения a, b содержат неизвестные; выражения c, d, e - не содержат. Антецедент упрощает сумму четвертых степеней с помощью нормализатора раскрывания скобок "стандплюс". Проверяется, что эта сумма тоже не содержит неизвестных. Уровень срабатывания равен 4.

10. Возведение в квадрат уравнения с тремя радикалами четвертой степени.
    $\forall_{abcdef}(a\sqrt[4]{b} + c\sqrt[4]{d} + e\sqrt[4]{f} = 0 \leftrightarrow a^2\sqrt{b} + c^2\sqrt{d} + 2ac\sqrt[4]{b}\sqrt[4]{d} - e^2\sqrt{f} = 0 \ \& \ ef(a\sqrt[4]{b} + c\sqrt[4]{d}) \leq 0)$

    Выражения b, d, f содержат неизвестные. Уровень срабатывания равен 6.

11. Устранение радикала путем преобразования подкоренного выражения с помощью дополнительного уравнения. В некоторых случаях неизвестное выражение под радикалом может быть преобразовано к виду полного квадрата с помощью других уравнений задачи:

    $\forall_{abcdepq}(\neg(c = 0) \ \& \ e - d + bc = mp^2 \ \& \ ac + d = e \rightarrow \sqrt{a + b} = \sqrt{m/c}|p|)$

    Преобразуется условие задачи на описание, имеющей более одной неизвестной. Выражение a содержит неизвестные и входит в другое уравнение $ac + d = e$, где c известно. Второй антецедент обрабатывает нормализатором "видумножение" выражение $e - d + bc$, равное $ac + bc$. Произведение всех известных множителей результата идентифицируется с m и проверяется, что произведение остальных множителей представляет собой полный квадрат. Уровень срабатывания равен 4.

12. Попытка усмотрения полного квадрата в неизвестной тригонометрической сумме под радикалом.

    $\forall_{abcdef}(e = b + c \rightarrow a\sqrt{b + c} + d = f \leftrightarrow a\sqrt{e} + d = f)$

    Преобразуется условие задачи на описание. Сумма под радикалом содержит тригонометрическую операцию с неизвестным аргументом. Антецедент обрабатывает ее нормализатором "видумножение". Проверяется, что результатом служит выражение, идентифицируемое с полным квадратом либо отношением двух полных квадратов. Уровень срабатывания равен 2.

Показательные уравнения

1. Логарифмирование показательных уравнений. Созданы несколько различных приемов, применяемых в зависимости от группировки членов в преобразуемом уравнении и отличающихся способом выбора основания логарифма. Начнем со

случая, когда члены уравнения не требуют перегруппировки перед логарифмированием:

$\forall_{abcd}(0 < a \ \& \ 0 < b \ \& \ 0 < c \ \& \ \neg(b - 1 = 0) \to ab^d = c \leftrightarrow d - \log_b c = -\log_b a)$

Прием применяется к условию задачи на описание либо к посылке задачи на исследование. Уровень срабатывания его равен 0, поэтому накладывается целый ряд дополнительных требований на целесообразность срабатывания. При их нарушении будут применяться приемы с несколько большими уровнями срабатывания. Задача должна иметь единственную неизвестную. Показатель степени d содержит неизвестную, а основание b - не содержит. Правая часть c может содержать неизвестные, но в любом случае не должна иметь заголовка "плюс". Каждый неизвестный сомножитель выражений a, c должен иметь вид степени с известным основанием. b - либо десятичная константа, либо самое короткое из оснований неизвестных степенных множителей выражений a, c. Отсутствует комментарий "общаястепень", блокирующий логарифмирование показательных уравнений. Обычно этот комментарий вводится при обратном действии - потенцировании логарифмических уравнений. Прием вводит комментарий "логарифм", блокирующий последнее.

$\forall_{abcd}(0 < a \ \& \ 0 < b \ \& \ \neg(b - 1 = 0) \to a^c = d \leftrightarrow c \log_b a = \log_b d \ \& \ 0 < d)$

Указатель "контекст(условие(x5)разряд(x5 логарифм x6)равно(x2 первыйоперанд(x6)))" определяет идентификацию основания b, по которому будет выполняться логарифмирование, через усмотрение в уравнениях задачи некоторого выражения $\log_b f$ с неизвестным f. Прием применяется к условию задачи на описание. Показатель степени c содержит неизвестные. Уровень срабатывания равен 1.

$\forall_{abcd}(0 < a \ \& \ \neg(a - 1 = 0) \ \& \ 0 < d \to da^b = c \leftrightarrow 0 < c \ \& \ \log_a d + b = \log_a c)$

Прием применяется к условию задачи на описание либо к посылке задачи на исследование. Выражение b содержит неизвестные; выражения a, c - не содержат. Если решается задача на исследование, то уравнение не должно иметь неизвестных, встречающихся в посылках вида "целое(...)", "натуральное(...)". Уровень срабатывания равен 1.

$\forall_{abcd}(0 < a \ \& \ 0 < b \to ba^c = d \leftrightarrow a - 1 = 0 \ \& \ b = d \ \vee \ \neg(a - 1 = 0) \ \& \ 0 < d \ \& \ \log_a b + c = \log_a d)$

Прием применяется к условию задачи на описание. Показатель степени c содержит неизвестные, выражение d - не содержит. Если выражение b известно, то прием применяется даже при наличии комментария "общаястепень", запрещающего логарифмрование показательных уравнений (предыдущие приемы этим комментарием блокировались). В этом случае получается уравнение, заведомо более простое, чем исходное. Уровень срабатывания равен 2. Несколько вырожденных случаев вынесены в отдельные приемы:

$\forall_{abcd}(0 < a \to ba^c = ba^d \leftrightarrow b = 0 \ \vee \ a - 1 = 0 \ \vee \ c = d)$

Преобразуется условие задачи на описание либо посылка задачи на исследование. Либо c, либо d не известно. Уровень срабатывания равен 0. Следующие два приема усматривают замаскированную первую степень суммы $a + b$, распределенной между различными частями равенства:

$\forall_{abc}((a + b)^c - a = b \leftrightarrow a + b = 1 \ \vee \ c = 1)$

$\forall_{abc}(a-(a+b)^c=-b \leftrightarrow a+b=1 \ \vee \ c=1)$

Преобразуется уравнение задачи на описание. a, c содержат неизвестные. Уровень срабатывания равен 1.

$\forall_{ax}(\neg(a=0) \to a^x=1 \leftrightarrow x=0 \vee a=1 \vee a=-1 \vee x-\mathrm{rational} \ \& \ \text{числитель}(x)-\mathrm{even})$

Прием применяется к содержащему неизвестные условию задачи на описание. Уровень срабатывания равен 4. Перейдем к уравнениям, все ненулевые члены которых сгруппированы в левой части. Так hпроизойдет, если все эти члены содержат неизвестные. Начнем с приема, срабатывающего на уровне 0 и являющегося двойником рассмотренного выше приема с ненулевой правой частью:

$\forall_{abcd}(0<a \ \& \ 0<b \ \& \ c<0 \ \& \ \neg(b-1=0) \to ab^d+c=0 \leftrightarrow d-\log_b(-c)=-\log_b a)$

Прием применяется к условию задачи на описание либо к посылке задачи на исследование. Показатель степени d содержит неизвестные, выражение b - не содержит. Выражение c не имеет вида суммы. Каждый неизвестный сомножитель выражений a, c представляет собой степень с известным основанием. b - либо десятичная константа, либо самое короткое из оснований неизвестных степенных множителей выражений a, c. Отсутствует комментарий "общаястепень".

$\forall_{abcd}(0<a \to ba^c+d=0 \leftrightarrow 0<b \ \& \ d<0 \ \&$
$log_2 b+c\log_2 a-\log_2(-d)=0 \ \vee \ b<0 \ \& \ 0<d \ \& \ \log_2(-b)+c\log_2 a-\log_2 d=0)$

Прием применяется к условию задачи на описание. Выражения a, c должны содержать неизвестные, а d не должно иметь вида суммы. Логарифмирование выполняется по основанию 2. Проверяется, что выражения a, b, d не имеют сомножителей, основание степени которых содержит неизвестный логарифм. Отсутствует комментарий "общаястепень". Указатель "попытказамены" делает прием "осторожным" - он не изменяет условия текущей задачи, а предпринимает попытку решения вспомогательной задачи, полученной указанной заменой условия. Если ее удается решить, то выдается ответ, иначе продолжается сканирование текущей задачи. Уровень срабатывания равен 2. Еще одна версия приема для той же теоремы создана на уровне 4. Она отличается лишь тем, что a известно, а c не известно.

$\forall_{abcde}(0<a \ \& \ 0<b \ \& \ \neg(b-1=0) \to ca^d+e=0 \leftrightarrow c=0 \ \& \ e=0 \ \vee \ 0<c \ \& \ e<0 \ \& \ \log_b c+d\log_b a-\log_b(-e)=0 \vee c<0 \ \& \ 0<e \ \& \ \log_b(-c)+d\log_b a-\log_b e=0)$

Этот прием аналогичен предыдущему, однако не требуется, чтобы a и c были неизвестны. Основание b, по которому будет выполняться логарифмирование, выбирается по неизвестному выражению вида $\log_b a$, входящему в какое-либо уравнение задачи. Уровень срабатывания равен 2. В заключение приведем два приема, ориентированных на вырожденные случаи:

$\forall_{abcd}(0<a \to ba^c-ba^d=0 \leftrightarrow b=0 \ \vee \ a-1=0 \ \vee \ c=d)$

Прием применяется к условию задачи на описание либо посылке задачи на исследование. Либо c, либо d содержит неизвестные. Уровень срабатывания равен 0.

$\forall_{abcde}(0<b \to ab^c+db^e=0 \leftrightarrow ab^{c-e}+d=0)$

Прием применяется к условию задачи на описание. Выражение c, а также хотя бы одно из выражений b, e содержит неизвестные. a, d известны. Уровень срабатывания равен 0.

2. Выражение степени разности через степень суммы, если разность квадратов известна. Предпринимается стандартизация оснований неизвестных степеней:

   $\forall_{abcd}(0 < a + b \ \& \ 0 < a - b \ \& \ c = a^2 - b^2 \rightarrow (a - b)^d = c^d(a + b)^{-d})$

   Преобразуется условие задачи на описание. Показатель степени d содержит неизвестные. Кроме выражения $(a - b)^d$, преобразуемое условие должно также содержать выражение вида $(a + b)^e$. Третий антецедент вычисляет разность квадратов c, используя нормализатор раскрывания скобок "стандплюс". Проверяется, что c не содержит неизвестных и либо $= 1$, либо c^d встречается в рассматриваемом условии. Если c имеет заголовок "минус", прием блокируется. Уровень срабатывания равен 2.

3. Разложения натурального основания неизвестной степени в произведение двух натуральных множителей.

   $\forall_{abcde}(b = ac \rightarrow a^d b^e = a^{d+e} c^e)$

   Выражения a, b идентифицируются с натуральными константами. Оба показателя степени d, e содержат неизвестные текущей задачи на описание. Антецедент выделен указателем "программа" - он делит b на a. После нескольких применений приема возникнут два константных основания, не делящиеся друг на друга. Уровень срабатываня равен 0.

4. Сведение к квадратному уравнению показательных уравнений. Несколько приемов усматривают квадратное уравнение относительно неизвестного показательного выражения и обращаются к несложному нормализатору "квадруравн", позволяющему сразу получить явные выражения для корней уравнения.

   $$\forall_{abcdefg}\left(\frac{a}{b^{c+d}} + \frac{e}{b^{2c+f}} = g \leftrightarrow \frac{a}{b^d}\frac{1}{b^c} + \frac{e}{b^f}\left(\frac{1}{b^c}\right)^2 = g\right)$$

   Преобразуется условие задачи на описание. Выражение c содержит неизвестные; выражения a, e, g - не содержат. d и f идентифицируются как суммы всех известных слагаемых соответствующих показателей степени. Если b содержит неизвестные, то d, f равны 0. Здесь и в дальнейших приемах подраздела заменяющий терм обрабатывается нормализатором "квадруравн". Уровень срабатывания равен 1.

   $\forall_{abcdefghijk}(0 < a \ \& \ b - 2c = f \ \& \ d = ae \ \& \ g - 2c = h \rightarrow ie^b + jd^c + ka^g = 0 \leftrightarrow ie^f((e/a)^c)^2 + j(e/a)^c = -ka^h)$

   Показатели степени b, c, g содержат неизвестные, коэффициенты i, j, k - не содержат. Третий антецедент проверяет, что основание степени d является произведением двух других оснований a, e. Второй и четвертый антецеденты определяют остаточные показатели степени f, h, возникающие после выделения квадратного трехчлена относительно $(e/a)^c$. Эти показатели должны не содержать неизвестных. Проверяется, что либо a известно, либо $h = 0$, а также что либо

e известно, либо $f = 0$. Таким образом обеспечивается независимость коэффициентов квадратного трехчлена от неизвестных. Уровень срабатывания равен 2.

$\forall_{abcdefghijk}(0 < a \;\&\; b - 2c = f \;\&\; g - 2d = e \rightarrow hk^b + ik^c a^d + ja^g = 0 \leftrightarrow hk^f(k^c/a^d)^2 + i(k^c/a^d) = -ja^e)$

Показатели степени b, c, d, g содержат неизвестные, коэффициенты h, i, j и остаточные показатели степени e, f - не содержат. Проверяется, что либо a известно, либо $e = 0$, а также что либо k известно, либо $f = 0$. Уровень срабатывания равен 2.

$\forall_{abcdpqrh}(0 < a \;\&\; h = p - 2q + r \rightarrow ba^p + ca^q + da^r = 0 \leftrightarrow ba^h(a^{q-r})^2 + ca^{q-r} = -d)$

Показатели степени p, q, r содержат неизвестные, коэффициенты b, c, d и остаточный показатель степени h известны. Либо a известно, либо $h = 0$. Уровень срабатывания равен 2.

5. Представление произведения неизвестных степеней с известными основаниями в виде степени произведения этих оснований, домноженной на известный коэффициент.

   $\forall_{abcde}(0 \le a \;\&\; 0 \le d \rightarrow a^{b+c}d^{b+e} = a^c d^e (ad)^b)$

   Преобразуется условие задачи на описание либо на преобразование. Выражения c, e идентифицируются как суммы всех известных слагаемых соответствующих показателей степени. Общий остаток b должен содержать неизвестные. Основания степени a, d не содержат неизвестных. Переменные заменяемого выражения не связаны внешними кванторами и описателями. Чтобы блокировать рассмотрение симметричных случаев, введен фильтр "постпозиция(фикс(0 1 2)фикс(0 1 1))". d^{b+e} будет идентифицироваться с сомножителем, идущим после того сомножителя, с которым идентифицировано a^{b+c}. Уровень срабатывания равен 1. Аналогичный прием создан для редких случаев неизвестного рационального показателя степени, имеющего нечетный знаменатель:

   $\forall_{abc}(c - \text{rational} \;\&\; \neg(\text{знаменатель}(c) - \text{even}) \rightarrow a^c b^c = (ab)^c)$

   Здесь c содержит неизвестные; a, b - известны. Уровень срабатывания равен 1.

6. Усмотрение в левой части уравнения произведения двух показательных двучленов.

   $\forall abcdepqr(cda^{r+q}/bea^p = -1 \;\&\; 0 < a \rightarrow ba^p + ca^q + da^r = e \leftrightarrow (ca^q - e)(da^r - e) = 0)$

   Преобразуется условие задачи на описание. Показатели степени p, q, r содержат неизвестные; основание степени a и коэффициенты b, c, d известны. Правая часть e ненулевая. Первый антецедент упрощает левую часть равенства, используя нормализаторы общей стандартизации, и проверяет, что результат равен -1. Уровень срабатывания равен 2.

7. Представление неизвестной степени с натуральным основанием в виде произведения двух неизвестных степеней с натуральными основаниями, уже имеющихся в уравнении.

   $\forall_{abcdefghpqmnrs}(e = b^n s \;\&\; s = h^m \rightarrow ab^c + de^f + gh^p + q = r \leftrightarrow ab^c + db^{nf}h^{mf} + gh^p + q = r)$

Преобразуется уравнение задачи на описание. Основания степеней b, e, h - натуральные константы. Показатели степени c, f, p содержат неизвестные. Антецеденты выделены указателем "программа". Первый из них находит наибольшое натуральное n, при котором e делится на b^n. Второй проверяет, что частное является натуральной степенью числа h. В итоге основание степени e заменяется на $b^n h^m$. Уровень срабатывания равен 3.

8. Существование решения простейшего показательного уравнения.

   $\forall_{ab}(0 < a \to \exists_x(x - \text{число} \ \& \ a^x = b) \leftrightarrow 0 < b)$

   Прием имеет заголовок "связка". Он применяется к задачам на описание, имеющим несущественную неизвестную x, не входящую в a, b. Условиями задачи, содержащими данную неизвестную, являются только "x – число" и $a^x = b$. Уровень срабатывания равен 0.

9. Степень минус единицы. Для решения простейшего показательного уравнения с основанием -1 созданы следуюющие два приема:

   $\forall_n(n - \text{целое} \to (-1)^n = -1 \leftrightarrow \neg(n - \text{even}))$

   $\forall_n(n - \text{целое} \to (-1)^n = 1 \leftrightarrow n - \text{even})$

   Преобразуется условие задачи на описание. Выражение n содержит неизвестные. Уровень срабатывания равен 1.

Разбор случаев по знаку неизвестного сомножителя основания степени

$\forall_{ab}(0 \leq ab \to 0 < a \ \& \ 0 < b \ \vee \ a < 0 \ \& \ b < 0 \ \vee \ a = 0 \ \& \ b = 0)$

Прием имеет заголовок "выводусловия". Указатель "контрольвывода(степень(умножение(x1 x2)x3))" инициирует попытку применения приема при усмотрении степени $(ab)^c$ внутри уравнения задачи на описание. a - некоторый неизвестный сомножитель основания этой степени. Правая часть уравнения нулевая, причем каждое слагаемое левой части имеет обобщенный множитель (достижимый из корня слагаемого через операции "умножение", "минус", "дробь" и основания степени), являющийся сомножителем выражения ab. Уравнение также не должно иметь неизвестных тригонометрических выражений. Новое дизъюнктивное условие снабжается комментарием "разборслучаев", форсирующим применение разбора случаев. Уровень срабатывания равен 2.

Разбор случаев по знаку неизвестного знаменателя дроби, являющейся основанием степени

$\forall_b(\neg(b = 0) \to 0 < b \ \vee \ b < 0)$

Прием имеет заголовок "выводусловия". Указатель "контрольвывода(степень(дробь(x1 x2)x3))" определяет инициализацию попытки его применения при усмотрении степени $(a/b)^c$ внутри некоторого уравнения задачи на описание. Для срабатывания приема необходимо выполнение следующих дополнительных требований:

1. Задача имеет единственную неизвестную.
2. Выражение b содержит неизвестную, причем неравенства $0 < b$, $b < 0$ из контекста не усматриваются.

3. Либо правая часть рассматриваемого уравнения равна 0, либо выделенная степень $(a/b)^c$ является сомножителем произведения, делящегося также на b.
4. Каждое слагаемое левой части уравнения должно иметь своим обобщенным множителем a либо b.
5. Задача не должна иметь условия, снабженного комментарием "разборслучаев".
6. Выделенная степень не является сомножителем левой части уравнения, правая часть которого равна 0.

При решении дифференциальных уравнений прием блокируется. Новое дизъюнктивное условие снабжается комментарием "разборслучаев". Уровень срабатывания равен 0.

Разбор случаев по четности целочисленного показателя степени с отрицательным основанием

$\forall_n(n - \text{целое} \to n - \text{even} \ \vee \ \neg(n - \text{even}))$

Указатель "контрольвывода(степень(минус(x1)x14))" инициирует попытку применения приема при усмотрении выражения $(-a)^n$ в условии задачи на описание. Выражение n содержит неизвестные. Уровень срабатывания равен 3. Для тригонометрических уравнений создана еще одна версия приема, срабатывающая на уровне 1:

$\forall_n(n - \text{even} \ \vee \ \neg(n - \text{even}))$

Здесь n - переменная, причем имеется условие вида "целое(n)". Преобразуется уравнение задачи на описание, содержащее неизвестное тригонометрическое выражение. Рассматриваемая степень $(-a)^n$ не расположена под тригонометрической операцией.

Равенство нулю линейной комбинации двух неизвестных степеней, показатели которых имеют общий известный множитель

Выполняется переход к уравнению, полученному из исходного возведением в степень, обратную общему множителю показателей:

$\forall_{abcde}(a - \text{rational} \ \& \ \text{знаменатель}(a) - \text{even} \ \& \ 0 \leq b \ \& \ 0 \leq d \ \& \ 0 \leq e \to bd^a + ce^a = 0 \leftrightarrow c \leq 0 \ \& \ b^{1/a}d - (-c)^{1/a}e = 0 \ \vee \ e = 0 \ \& \ bd^a = 0)$

Преобразуется условие задачи на описание либо посылка задачи на исследование. Указатели "перечень(x2 известно(x2))", "перечень(x3 известно(x3))" идентифицируют коэффициенты b, c как произведения всех известных сомножителей соответствующих слагаемых. Указатель "общаястепень(общаястепень x1 степень(x4 x1)степень(x5 x1))" определяет идентификацию остатков слагаемых в виде степенных выражений d^a, e^a при помощи вспомогательной процедуры "общаястепень". Эта процедура рассматривает каждый сомножитель остатков, убеждается в том, что он имеет вид степени, и находит наибольший общий делитель a показателей всех таких степеней. В случае дробных показателей a тоже может иметь вид дроби. Выражения d, e должны содержать неизвестные (вырожденные остатки не допускаются). Введен ускоряющий фильтр, проверяющий, что каждый неизвестный сомножитель каждого слагаемого левой части уравнения имеет вид степени с неизвестным основанием. Если a равно 1/2, то прием блокируется (в этой ситуации применяются приемы возведения уравнения в квадрат). Уровень срабатывания равен 2.

$\forall_{abcde}(e - \text{rational}\ \&\ \neg(\text{числитель}(e) - \text{even})\ \&\ \neg(\text{знаменатель}(e) - \text{even}) \to ab^e + cd^e = 0 \leftrightarrow a^{1/e}b + c^{1/e}d = 0)$

$\forall_{abcde}(0 \leq a\ \&\ c \leq 0\ \&\ e - \text{rational}\ \&\ \text{числитель}(e) - \text{even}\ \&\ \neg(e = 0) \to ab^e + cd^e = 0 \leftrightarrow a^{1/e}b + (-c)^{1/e}d = 0\ \vee\ a^{1/e}b - (-c)^{1/e}d = 0)$

Аналогично предыдущему, но уровни срабатывания равны 1. При этом в последнем приеме требуется, чтобы либо b, либо d было суммой. Если общий множитель показателей не является рациональным числом, то применяется следующий прием:

$\forall_{abcde}(0 \leq b\ \&\ 0 \leq d\ \&\ 0 \leq a\ \&\ 0 \leq e\ \&\ 0 < c \to ab^c - ed^c = 0 \leftrightarrow a^{1/c}b - e^{1/c}d = 0)$

Преобразуется условие задачи на описание. Выражения a, c, e известны; b, d - содержат неизвестные. Уровень срабатывания равен 3.

Занесение минуса в основание степени при решении уравнения

Выполняется занесение минуса в основание степени либо перестановка частей разности, являющейся таким основанием. В обоих случаях обеспечивается переход к уже имевшемуся в уравнении другому основанию:

$\forall_{abc}(a - \text{rational}\ \&\ \neg(\text{числитель}(a) - \text{even})\ \&\ \neg(\text{знаменатель}(a) - \text{even}) \to (b - c)^a = -(c - b)^a)$

$\forall_{ab}(b - \text{rational}\ \&\ \neg(\text{числитель}(b) - \text{even})\ \&\ \neg(\text{знаменатель}(b) - \text{even}) \to a^b = -(-a)^b)$

Преобразуется уравнение задачи на описание. Основание рассматриваемой степени содержит неизвестные. В первом случае уравнение содержит также степень вида $(c - b)^p$, во втором - степень $(-a)^p$; p - дробная константа с четным знаменателем. Уровень срабатывания равен 2.

Однородное уравнение второй степени

$\forall_{abcdxy}(d = b^2 - 4ac \to ax^2 + bxy + cy^2 = 0 \leftrightarrow \neg(a = 0)\ \&\ 0 \leq d\ \&\ (x = y(sqrtd - b)/2a\ \vee\ x = -y(\sqrt{d} + b)/2a)\ \vee\ a = 0\ \&\ y(bx + cy) = 0\ \vee\ \neg(a = 0)\ \&\ d < 0\ \&\ x = 0\ \&\ y = 0)$

Преобразуется условие задачи на описание. Выражения x, y содержат неизвестные; коэффициенты a, b, c известны. В случае дифференциальных уравнений прием блокируется, так как там алгебраические преобразования выполняются другими приемами. Если задача имеет единственную неизвестную, уровень срабатывания равен 3, иначе он равен 1. В случае задач на исследование, имеющих цель "известно", используется несколько упрощенный прием:

$\forall_{abcdxy}(d = b^2 - 4ac\ \&\ \neg(a = 0) \to ax^2 + bxy + cy^2 = 0 \leftrightarrow 0 \leq d\ \&\ (x = y(sqrtd - b)/2a\ \vee\ x = -y(\sqrt{d} + b)/2a)\ \vee\ d < 0\ \&\ x = 0\ \&\ y = 0)$

Созданы две версии. Первая срабатывает на уровне 4 и требует, чтобы уравнение не содержало невырожденных (отличных от переменных и констант) числовых атомов. Вторая лишена этого ограничения и срабатывает на уровне 8.

Выражение одной неизвестной через другую, если известно произведение их степеней

В двух специальных случаях форсируется выражение одной неизвестной задачи на описание через другую. Теорема приема имеет вид:

$\forall_{abcdef}(ab^c d^e = f \leftrightarrow ab^c d^e = f)$

Переменные b, d идентифицируются с неизвестными; выражения a, c, e, f известны. Заменяющий терм обрабатывается вспомогательной задачей на описание, разрешающей уравнение $ab^c d^e = f$ относительно b. Таким образом подготавливается последующее исключение неизвестной b. Заметим, что заменяющее утверждение может иметь вид дизъюнкции (например, если для a и f допустимы нулевые значения). Срабатывание приема возможно в двух случаях: либо каждое уравнение задачи имеет неизвестными множителями своих частей только степени неизвестных, либо существует уравнение вида $pb^x + qd^y = 0$, где хотя бы один из показателей x, y содержит неизвестные. В первом случае уровень срабатывания равен 3, во втором - 1.

Группировка под общий неизвестный показатель степеней с известными основаниями

Такая группировка является преобразованием общей стандартизации уравнений. Она обеспечивается следующими приемами.

$\forall_{abc}(0 \leq a \ \& \ 0 < b \rightarrow (a/b)^c = a^c/b^c)$

Преобразование выполняется справа налево. Дробь находится в условии задачи на описание либо на преобразование. Показатель степени c содержит неизвестные, а выражения a, b не содержат. Преобразуемое выражение не должно быть расположено внутри конечной суммы. Уровень срабатывания равен 1.

$\forall_{abcdefg}(c - \text{число} \ \& \ g - \text{rational} \ \& \ \neg(\text{знаменатель}(g) - \text{even}) \ \& \ e - \text{rational} \ \& \ \neg(\text{знаменатель}(e) - \text{even}) \ \& \ \neg(c = 0) \ \& \ \neg(\text{знаменатель}(a) - \text{even}) \ \& \ a - \text{rational} \ \& \ \neg(f = 0) \ \& \ f - \text{число} \rightarrow db^{ae}/fc^{ag} = d(b^e/c^g)^a/f)$

Прием применяется слева направо. Выражение a содержит неизвестные, выражения b, c, d, e, f, g - не содержат. Уровень срабатывания равен 1. Переменные d, e, f, g могут принимать вырожденные единичные значения.

$\forall_{abcd}(0 \leq a \ \& \ 0 \leq b \rightarrow a^c b^{cd} = (ab^d)^c)$

Прием применяется в задачах на описание, преобразование либо исследование. Выражение c содержит неизвестные, выражения a, b, d - не содержат. Текущий терм задачи не имеет вне преобразуемого выражения вхождений степеней a^x, b^x с неизвестным показателем x. Переменные этого выражения не связаны внешними кванторами и описателями. Если задачи имеет цель (независит …), запрещающую вхождение в ответ заданных переменных, то они не встречаются в a, b, d. Уровень срабатывания равен 1.

$\forall_{abceghi}(\neg(\text{знаменатель}(c) - \text{even}) \ \& \ \neg(\text{знаменатель}(e) - \text{even}) \ \& \ \neg(\text{знаменатель}(g) - \text{even}) \ \& \ \neg(\text{знаменатель}(h) - \text{even}) \ \& \ \neg(\text{знаменатель}(i) - \text{even}) \ \& \ c - \text{rational} \ \& \ e - \text{rational} \ \& \ g - \text{rational} \ \& \ h - \text{rational} \ \& \ i - \text{rational} \rightarrow a^i b^h (a^e b^c)^g = a^{i+eg} b^{h+cg})$

Преобразование выполняется справа налево. Выражение g содержит неизвестные, выражения a, b, c, e - не содержат. В остальном прием аналогичен предыдущему. Так как требуется, чтобы g было рациональным, прием применяется в крайне редких ситуациях. Уровень срабатывания равен 1.

Группировка слагаемых с одинаковым неизвестным радикалом

$\forall_{abcdefgh}(a = fg \ \& \ e = fh \rightarrow ab^{c/d} + eb^{c/d} = f(g+h)b^{c/d})$

Преобразуемая сумма расположена в условии задачи на описание либо в посылке задачи на исследование. Она является левой частью уравнения либо (только в случае задач на описание) неравенства, не используемого для сопровождения по о.д.з. Выражение b содержит неизвестные. Коэффициенты a, e тоже могут содержать неизвестные, но не под тригонометрическими операциями. Антецеденты определяют произведение f общих множителей коэффициентов, а также остаточные коэффициенты g, h. Последние не должны иметь своими множителями степени с неизвестным основанием и дробным показателем. При решении дифференциальных уравнений, если g, h содержат производные, прием блокируется. Блокируется он также при решении планиметрических задач. Уровень срабатывания равен 1.

Два степенных уравнения с известными показателями

Для простейшей системы из двух степенных уравнений с двумя неизвестными a, b создан прием, сразу выписывающий ответ:

$$\forall_{abcdpqrsm}(0 < a \;\&\; 0 < b \;\&\; m = qr - ps \rightarrow a^p b^q = c \;\&\; a^r b^s = d \leftrightarrow \neg(m = 0) \;\&\; a = (d^q/c^s)^{1/m} \;\&\; b = (c^r/d^p)^{1/m} \vee m = 0 \;\&\; a^p b^q = c \;\&\; c^r = d^p \;\&\; (\neg(r = 0) \vee r = 0 \;\&\; b^s = d))$$

Прием имеет заголовок "заменаусловия(воройтерм)" - он заменяет сразу оба уравнения задачи на описание на одно новое утверждение. Выражения a, b содержат неизвестные (хотя сами могут не являться переменными), выражения p, q, r, s, c, d неизвестных не содержат. Число неизвестных задачи больше единицы. Уровень срабатывания равен 2.

Переход к однородному уравнению второй степени с двумя неизвестными путем рассмотрения линейной комбинации двух неоднородных

$$\forall_{abcdefghijk}(ai^2 + bij + cj^2 = dk \;\&\; \neg(d = 0) \rightarrow ei^2 + fij + gj^2 = hk \leftrightarrow (de - ha)i^2 + (df - hb)ij + (dg - hc)j^2 = 0)$$

Преобразуется условие задачи на описание либо посылка задачи на исследование. Первый антецедент идентифицируется с другим уравнением задачи. Число неизвестных более единицы. Выражения i, j содержат неизвестные, правые части уравнений ненулевые и неизвестных не содержат. Выражение k идентифицируется как произведение общих множителей правых частей, коэффициенты a, b, c, e, f, g - как произведения всех известных множителей соответствующих слагаемых. Указатели "подстановка(...)" обеспечивают идентификацию при обращении в ноль коэффициентов e, f, a, c. Заменяющий терм обрабатывается нормализатором "квадруравн", сразу выписывающим решение однородного уравнения. Уровень срабатывания равен 2. Для одного специального случая создан дополнительный прием, срабатывающий на уровне 1:

$$\forall_{abcdexy}(axy + bx^2 = ce \;\&\; ay^2 + bxy = de \leftrightarrow \neg(c = 0) \;\&\; y = dx/c \;\&\; (ad + bc)x^2 = c^2 e \;\vee\; c = 0 \;\&\; \neg(d = 0) \;\&\; x = 0 \;\&\; ay^2 = de \;\vee\; c = 0 \;\&\; d = 0 \;\&\; ay + bx = 0)$$

Прием имеет заголовок "заменаусловия(воройтерм)" и заменяет оба уравнения. Выражения x, y содержат неизвестные, выражения a, b, c известны. Правые части уравнений ненулевые.

Использование значения для модуля основания неизвестной степени с четным показателем

$\forall_{abm}(|a| = b \to a^m = b^m)$

Выражение a содержит неизвестные, а выражение b не содержит. m - целая четная константа. Уровень срабатывания равен 4.

Уравнения в задачах на исследование, имеющих цель "известно"

Напомним, что в сколь-нибудь сложных случаях система уравнений, сложившаяся в задаче на исследование, имеющей цель "известно", решается во вспомогательной задаче на описание. Все неизвестные такой системы должны быть числовыми переменными. Однако, для ряда простых часто встречающихся случаев создаются специальные приемы (иногда дублирующие приемы задач на описание), применяемые непосредственно в задаче на исследование. Перечислим такие приемы, связанные со степенными и показательными уравнениями:

1. Определение суммы квадратов, если известны сумма и произведение.

   $\forall_{abcdp}(a + b = c \,\&\, ab = d \to pa^2 + pb^2 = p(c^2 - 2d))$

   Прием выполняет тождественную замену. Выражения a, b содержат неизвестные, выражения c, d = не содержат. Уровень срабатывания равен 6.

2. Деление уравнений для устранения радикала.

   $\forall_{abcdemnk}(a/b\sqrt{c} = mk \,\&\, \neg(b = 0) \,\&\, \neg(m = 0) \to d/e\sqrt{c} = nk \leftrightarrow aen - bdm = 0)$

   Преобразуется посылка задачи на исследование либо на доказательство. Первый антецедент идентифицируется с другой посылкой. Выражение c содержит неизвестные; b, e, m, n - константные выражения. Уравнения не содержат невырожденных (отличных от переменных и констант) числовых атомов. Уровень обращения равен 6. Заметим, что в задачах на доказательство неизвестной считается любая переменная, не выделенная комментарием "известно". Некоторые из приводимых ниже приемов, как и данный прием, могут кроме задач на исследование применяться также в задачах на доказательство.

3. Извлечение корня из обеих частей уравнения.

   $\forall_{abcdn}(ab^n/e = cd^n/f \leftrightarrow (a/e)^{1/n}b = (c/f)^{1/n}d)$

   Выражения b, d содержат неизвестные; выражения a, c, e, f - не содержат. n - рациональная константа с нечетными числителем и знаменателем. Уровень срабатывания равен 5. На той же теореме создана еще одна версия приема, применяемая в тех случаях, когда b содержит невырожденный числовой атом, а a, c, e, f - не содержат.

   $\forall_{abcdn}(0 \leq ae \,\&\, 0 \leq cf \to ab^n/e = cd^n/f \leftrightarrow (a/e)^{1/n}b = (c/f)^{1/n}d \;\vee\; (a/e)^{1/n}b = -(c/f)^{1/n}d)$

   Аналогично предыдущему, но n - рациональная константа с четным числителем. Прием используется только в тех задачах, где рассматривается прямоугольная система координат. Уровень срабатывания равен 5.

   $\forall_{abcdn}(0 \leq a \,\&\, 0 \leq c \to ab^n - cd^n = 0 \leftrightarrow a^{1/n}b = c^{1/n}d \;\vee\; a^{1/n}b = -c^{1/n}d)$

Выражения b, d содержат неизвестные; выражения a, c - не содержат. n - рациональная константа с четным числителем. В планиметрии прием блокируется. Уровень срабатывания равен 5. Для случая неотрицательных оснований созданы отдельные приемы, срабатывающие на уровне 4:

$\forall_{abcdn}(0 \leq a \;\&\; 0 \leq c \;\&\; 0 \leq b \;\&\; 0 \leq d \to ab^n - cd^n = 0 \leftrightarrow a^{1/n}b = c^{1/n}d)$

$\forall_{abcdn}(0 \leq a \;\&\; 0 \leq c \;\&\; 0 \leq b \;\&\; 0 \leq d \to ab^n = cd^n \leftrightarrow a^{1/n}b = c^{1/n}d)$

Наконец, имеется прием, обрабатывающий соотношение пропорциональности квадратов двух неизвестных:

$\forall_{abxy}(0 < a \;\&\; 0 < x^2 + y^2 \to ax^2 + by^2 = 0 \leftrightarrow b \leq 0 \;\&\; (\sqrt{a}x + \sqrt{-b}y = 0 \vee \sqrt{a}x - \sqrt{-b}y = 0))$

Здесь x, y - неизвестные; a, b - выражения без неизвестных. Преобразованная посылка снабжается комментарием "разборслучаев". Уровень срабатывания равен 5.

4. Линейная комбинация уравнений для устранения общего радикала.

   $\forall_{abcdpqr}(\neg(b = 0) \;\&\; a + b\sqrt{c} = d \to p + q\sqrt{c} = r \leftrightarrow aq - bp = dq - br)$

   Второй антецедент идентифицируется со вторым уравнением. Выражение c неконстантное и имеет вид суммы. Преобразуемое уравнение не имеет невырожденных числовых атомов. Выражения b, d, q, r известны; каждое из выражений a, p линейно по всем своим параметрам. Прием применяется только в тех задачах, где упоминается прямоугольная система координат. Уровень срабатывания равен 3.

5. Возведение в квадрат уравнения с радикалом.

   $\forall_{abc}(0 \leq a \;\&\; 0 \leq b \;\&\; 0 \leq c \to a\sqrt{b} = c \leftrightarrow a^2b - c^2 = 0)$

   Выражения b, c содержат неизвестные, выражение a не содержит. Уравнение не имеет невырожденных числовых атомов. Число его переменных не более двух. Уровень срабатывания равен 6.

   $\forall_{abcd}(0 \leq d \to \sqrt{ba^2 + c} = d|a| \leftrightarrow c + (b - d^2)a^2 = 0)$

   Выражение a содержит неизвестные, выражения b, d - не содержат. Уровень срабатывания равен 4.

   $\forall_{abc}(0 \leq a \;\&\; 0 \leq b \;\&\; 0 \leq c \;\&\; 0 \leq d \to a\sqrt{b} = c\sqrt{d} \leftrightarrow a^2b - c^2d = 0)$

   Уравнение не имеет невырожденных числовых атомов. Выражения b, d содержат неизвестные, a, c - константы. Уровень срабатывания равен 4.

6. Возведение в квадрат для получения суммы попарных произведений. Прием возводит уравнение в квадрат (выводит следствие из него), если попарные произведения слагаемых его левой части могут представлять интерес для другого уравнения:

   $\forall_{abc}(a + b = c \to (a + b)^2 = c^2)$

   Левая часть уравнения $a + b = c$ представляет собой сумму переменных либо функциональных выражений $f(\ldots)$. Правая часть известна. Существует другое уравнение, содержащее сумму таких слагаемых, каждое из которых является произведением двух подвыражений первого уравнения. Левая часть результата обрабатывается нормализатором раскрывания скобок "стандплюс". Уровень срабатывания равен 3.

7. Использование посылки для сокращения неизвестных множителей.

   $\forall_{abcdefgpq}(a^b c = de \to pa^{b+f}/qd = pa^f e/qc)$

   Прием выполняет тождественную замену. Выражения b, d содержат неизвестные, выражения c, e, f - не содержат. Антецедент представляет собой посылку - соотношение пропорциональности для неизвестных выражений a^b, d. Уровень срабатывания равен 5.

8. Выражение квадрата неизвестной величины через ее известный модуль.

   $\forall_{ab}(|a| = b \to a^2 = b^2)$

   Прием выполняет тождественную замену. Антецедент идентифицируется с другой посылкой. Выражение a содержит неизвестные, b известно. Указатель "теквхожд(1)" определяет выбор точки привязки в антецеденте, т.е. сначала прием усматривает равенство неизвестного модуля известному выражению, а затем начинает искать квадрат выражения под модулем. Уровень срабатывания равен 2.

9. Использование посылки для произведения неизвестных величин.

   $\forall_{abcmn}(ab = c \;\&\; 0 \leq n - m \to a^m b^n = c^m b^{n-m})$

   Прием выполняет тождественную замену. Первый антецедент идентифицируется с другой посылкой. Выражения a, b содержат неизвестные, а также содержат невырожденные числовые атомы. c известно. m, n - натуральные константы. Уровень срабатывания равен 3.

10. Возведение в квадрат с сокращением неизвестных квадратов.

    $\forall_{abcd}(d + (a + b)^2 = (a + c)^2 \leftrightarrow d + 2a(b - c) + b^2 = c^2)$

    Выражение a содержит неизвестные, выражения b, c, d - не содержат. Уровень срабатывания равен 4.

Возведение в степень слагаемых двучленного уравнения для устранения дробной степени

$\forall_{abcmn}(a + bc^{m/n} = 0 \leftrightarrow a^n + b^n c^m = 0)$

Преобразуется условие задачи на описание. Выражение c содержит неизвестные. m, n - натуральные константы, причем n нечетно. Указатель "операнд(x1 фикс(0 1 1))" определяет идентификацию a с единственным слагаемым суммы, т.е. левая часть имеет всего два слагаемых. При решении дифференциальных уравнений прием блокируется. Уровень срабатывания равен 3.

$\forall_{abcmn}(0 \leq c \to a + bc^{m/n} = 0 \leftrightarrow a^n - b^n c^m = 0 \;\&\; abc \leq 0)$

Аналогично предыдущему, но n четно.

Попытка замены неизвестной для получения двучленного уравнения с дробной степенью

$\forall_{abcdmn}(\neg(a = 0) \;\&\; ax + b + c^{m/n} = d \to \exists_y(x = (y + d - b)/a \;\&\; y - \text{число}))$

Прием имеет заголовок "выводусловия". Он применяется к условию задачи на описание. x - неизвестная, не входящая в a, b. c содержит неизвестные, правая часть d

известна. m - константа 1 либо 2; n - натуральная константа, отличная от 2. Если задача имеет единственную неизвестную, то $n > 3$. Либо b, либо d отлично от 0. Новое условие снабжается комментариями "серия", "фильтрсерии". Тогда оно будет рассматриваться как параметрическое описание для неизвестной x, что вызовет переход к решению вспомогательной задачи относительно неизвестной y. Результатом станет преобразование уравнения к "двучленному" виду из предыдущего пункта.

Попытка применения тригонометрической замены

$\forall_x(\exists_y(x = \cos y \ \& \ 0 \le y \ \& \ y \le \pi \ \& \ y - \text{число}))$

Прием имеет заголовок "выводусловия" и применяется к условию задачи на описание. Указатель "контрольвывода(степень(плюс(1 минус(степень(x23 2)))дробь(1 2)))" определяет инициализацию применения приема с усмотрения внутри некоторого уравнения выражения $\sqrt{1-x^2}$, где x - неизвестная. Проверяется, что это выражение является сомножителем произведения, содержащего также x. Новое условие сопровождается комментариями "серия", "фильтрсерия", обеспечивающими переход к рассмотрению новой неизвестной y. Уровень срабатывания равен 2.

$\forall_{xy}(0 \le a \to x^2 + y^2 = a \leftrightarrow \exists_z(x = \sqrt{a}\sin z \ \& \ y = \sqrt{a}\cos z \ \& \ 0 \le z \ \& \ z < 2\pi \ \& \ z - \text{число}))$

Этот прием выполняет эквивалентную замену. x, y - неизвестные задачи на описание; a известно. Должно существовать еще одно уравнение, не линейное относительно неизвестных. Как и в предыдущем приеме, измененное условие сопровождается комментариями "серия", "фильтрсерии", обеспечивающими переход к новой неизвестной z. Уровень срабатывания равен 5.

Устранение дроби под радикалом при решении степенного уравнения

$\forall_{abcdx}(x = c\sqrt{a/b}/d \ \vee \ x = -c\sqrt{a/b}/d \leftrightarrow x = c\sqrt{ab}/bd \ \vee \ x = -c\sqrt{ab}/bd)$

Прием срабатывает на уровне 1 и применяется к условию задачи на описание.

Возведение в степень отрицания равенства для параметров при редактировании ответа

$\forall_{abcmn}(a + bc^{m/n} = 0 \leftrightarrow a^n + b^n c^m = 0)$

Равенство находится под отрицанием, внутри условия задачи на описание. Происходит редактирование ответа задачи. a, b - константные выражения; m, n - натуральные константы, причем n нечетное. Выражение c неконстантное. Прием применяется даже в тех случаях, когда утверждение используется для сопровождения по о.д.з. Уровень срабатывания равен 3.

$\forall_{abcmn}(0 \le c \ \& \ ab \le 0 \to a + bc^{m/n} = 0 \leftrightarrow b^n c^m - a^n = 0)$

Аналогично предыдущему, но n четное. Если утверждение используется для сопровождения по о.д.з., то прием блокируется.

Усмотрение в левой части уравнения четвертой степени однородного многочлена второй степени от неизвестной и квадратного двучлена

$\forall_{abcdefghijk}(-e = fj^2 \ \& \ a = fk^2 \ \& \ d = gj \ \& \ b = gk \ \& \ h = c - 2jkf \ \& \ \neg(e = 0) \to ai^4 + bi^3 + ci^2 + di = e \leftrightarrow \neg(i = 0) \ \& \ f((j + ki^2)/i)^2 + g(j + ki^2)/i + h = 0)$

Прием применяется к условию задачи на описание. Коэффициенты a, b, c, d, e суть целочисленные константы. Выражение i содержит неизвестные. Антецеденты выделены указателем "программа". Они определяют значения f, g, h и выполняют проверки, необходимые для возможности указанной в теореме группировки. Первый антецедент перечисляет делители числа $-e$ и отбирает те из них, которые представляют собой полный квадрат j^2. Произведение остальных делителей идентифицирует f. Второй антецедент находит a/f и проверяет, что оно является полным квадратом k^2. Третий антецедент идентифицирует g, четвертый выполняет дополнительную проверку, и пятый идентифицирует h. Указатели "подстановка" предусматривают возможность обращения в ноль коэффициентов b, c, d. Квадратное уравнение в заменяющем терме обрабатывается нормализатором "квадруравн", и прием сразу выписывает соотношения для возможных значений выражения $(j + ki^2)/i$. Уровень срабатывания равен 5.

Усмотрение в левой части уравнения четвертой степени квадратного трехчлена от квадратного двучлена

$$\forall_{ABCDEabpq}(pa^2 = A \ \& \ 2apb = B \ \& \ aq = C - pb^2 \ \& \ qb = D \rightarrow Ax^4 + Bx^3 + Cx^2 + Dx = E \leftrightarrow p(ax^2 + bx)^2 + q(ax^2 + bx) = E)$$

Прием аналогичен предыдущему. Коэффициенты A, B, C, D, E суть целочисленные константы; x - выражение с неизвестными. К заменяющему терму применяется нормализатор "квадруравн". Уровень срабатывания равен 7.

Уравнение третьей степени: случай неположительного дискриминанта

Для кубического уравнения с численными коэффициентами, имеющего неположительный дискриминант, выписывается формула с радикалами:

$$\forall_{abcdxpqr}(\neg(a = 0) \ \& \ p = (3ac - b^2)/3a^2 \ \& \ q = (2b^3 - 9abc - 27a^2d)/27a^3 \ \& \ r = q^2/4 + p^3/27 \ \& \ 0 < r \rightarrow ax^3 + bx^2 + cx = d \leftrightarrow x = \sqrt[3]{-q/2 + \sqrt{r}} + \sqrt[3]{-q/2 - \sqrt{r}} - b/3a)$$

$$\forall_{abcdxpqrA}(\neg(a = 0) \ \& \ p = (3ac - b^2)/3a^2 \ \& \ q = (2b^3 - 9abc - 27a^2d)/27a^3 \ \& \ r = q^2/4 + p^3/27 \ \& \ r = 0 \ \& \ A = -\sqrt[3]{4q} \rightarrow ax^3 + bx^2 + cx = d \leftrightarrow x = A \ \vee \ x = -A/2)$$

Преобразуется условие задачи на описание. Коэффициенты a, b, c, d - десятичные константы, выражение x содержит неизвестные. Второй, третий и четвертый антецеденты выделены указателем "идентификатор". Они используют нормализаторы общей стандартизации, причем q, r дополнительно упрощаются вспомогательными задачами на преобразование. Указатели "подстановка" допускают обращение в 0 коэффициентов b, c. Уровень срабатывания равен 7.

Уравнение третьей степени: тригонометрическое решение в случае положительного дискриминанта

$$\forall_{abcdxhmnpqr}(\neg(a = 0) \ \& \ p = (3ac - b^2)/3a^2 \ \& \ q = (2b^3 - 9abc - 27a^2d)/27a^3 \ \& \ r = q^2/4 + p^3/27 \ \& \ r < 0 \ \& \ m = -p/3 \ \& \ h = \pi - \arccos(q/2\sqrt{m^3}) \ \& \ n = \sqrt{m} \rightarrow ax^3 + bx^2 + cx = d \leftrightarrow x = 2n\cos(h/3) \ \vee \ x = -2n\cos(h/3 - \pi/3) \ \vee \ x = -2n\cos(h/3 + \pi/3))$$

Первые четыре антецедента такие же, как в предыдущем пункте. Эта часть программ приемов общая - компилятор склеивает идентичные начальные отрезки, так что получение r обеспечивается без дополнительных вычислительных затрат.

Специальный случай уравнения третьей степени

Прием усматривает в левой части уравнения выражение, пропорциональное кубу суммы:

$\forall_{ABCDEbc}(C = cB \;\&\; BE = 3bc \;\&\; D = -bc^3 \rightarrow Ax^3 + Bx^2 + Cx = D/E \leftrightarrow x(\sqrt[3]{b - AE} - \sqrt[3]{b}) = c\sqrt[3]{b})$

Преобразуется условие задачи на описание. Выражение x содержит неизвестные, выражения A, B, C, D, E - не содержат. Для b, c, A, B, C, E допускается вырожденное единичное значение. Уровень срабатывания равен 3.

Попытка выделения полного квадрата

Прием усматривает квадрат суммы неизвестных выражений a, b, предпринимает попытку выразить левую часть уравнения через $a+b$ и известные величины, после чего решает уравнение относительно $a + b$:

$\forall_{abcdefyA}(d - cb^2 = f(a+b) \;\&\; (cy^2 + f(y) = e \;\&\; y - \text{число}) = A(y) \rightarrow 2abc + ca^2 + d = e \leftrightarrow A(a+b))$

Преобразуется условие задачи на описание, имеющей единственную неизвестную x. Выражения a, b, d содержат неизвестные, выражения c, e известны. Антецеденты выделены указателем "идентификатор". После идентификации двух слагаемых $2abc, ca^2$ и остаточной суммы d первый антецедент обрабатывает выражение $d - cb^2$ с помощью вспомогательной задачи на упрощение. Попытка представить этот результат как некоторое $f(a + b)$, у которого все вхождения неизвестной x расположены только внутри $a + b$, обеспечивается указателем "новаргумент(x6 x23 извлечение)". Нормализатор "извлечение" выполняет необходимые тождественные преобразования. Второй антецедент составляет уравнение $cy^2 + f(y) = e$ относительно новой неизвестной y и решает его, обращаясь к вспомогательной задаче на описание. В найденный ответ $A(y)$ подставляется сумма $a + b$. Уровень обращения равен 6.

Разложение на множители квадратного трехчлена для повторной попытки выделения новых неизвестных

Если уравнение содержит произведение квадратного трехчлена от неизвестной на неизвестную сумму, то предпринимаются попытка доразложения этого произведения и повторная попытка перехода к новым неизвестным. Например, при получении произведения $x(x + 1)(x + 2)(x + 3)$ становится возможен ввод новой неизвестной $y = x^2 + 3x$ и переход к квадратному трехчлену $y(y + 2)$. Теорема приема имеет следующий вид:

$\forall_{abcdex}(d(ax^2 + bx = c) = e \rightarrow d(ax^2 + bx + c) = e)$

Преобразуется подвыражение условия задачи на описание, имеющего единственную неизвестную x. Коэффициенты a, b, c известны, выражение d имеет своим сомножителем неизвестную сумму. Антецедент обрабатывает свою левую часть нормализатором "видумножение". Проверяется, что число неизвестных сомножителей после попытки доразложения увеличивается. Указатель "удалениезамечания(комментарий(новыенеизвестные 2))" удаляет комментарий (новыенеизвестные 2), блокировавший повторную попытку перехода к новым неизвестным. Уровень срабатывания равен 6.

Коррекция сопровождения по о.д.з.

Если в задаче по каким-либо причинам возникло нарушение сопровождения по о.д.з. и его желательно восстановить, то вводится комментарий "коррекцияодз", активирующий восстанавливающие приемы. Один из таких приемов срабатывает, если усматривается степень с дробным показателем, имеющим четный знаменатель. Его теорема имеет следующий вид:

$\forall_a (0 \leq a)$

Указатель "контрольвывода(степень(x1 x2))" активирует прием при усмотрении в условии задачи на описание степенного выражения a^b, где b - простая дробь с четным знаменателем. Прием имеет заголовок "выводусловия". Проверяется, что неравенство $0 \leq a$ не очевидно из контекста и что a не константа. Задача должна иметь комментарий "контрольодз". Рассматриваемое вхождение степенного выражения не должно располагаться внутри квантора, описателя, условного выражения либо логичексих связок "и", "или". Указатель "контрольодз" обеспечивает восстановление комментариев (сопровождение ...) для подтермов текущего условия задачи. Уровень срабатывания равен 1.

Выдача промежуточного результата при нахождении числа корней

Если преобразуется условие принадлежности некоторому классу, причем представляет интерес лишь нахождение мощности данного класса, то используется ускоренная выдача ответа. Приведем простейший прием такого вида:

$ax^2 = b \;\&\; x -$ число

Заголовок приема - "ответзадачи". Конъюнктивные члены теоремы приема суть все содержащие неизвестную x условия задачи на описание, имеющей цель "мощность". a, b известны. Прием обрывает решение задачи, не доводя его до явного определения значения x, так как и без этого во внешнем контексте можно будет сразу найти мощность класса. Создан еще один аналогичный прием:

$ax^2 + bx = c \;\&\; x -$ число

Уровни срабатывания обоих приемов равны 0.

Усмотрение неположительного дискриминанта квадратного уравнения с неизвестными коэффициентами

Если усматривается неположительность дискриминанта, то этот дискриминат равен 0, и уравнение распадается на два уравнения:

$\forall_{abcdex}(d = b^2 - 4a(c - e) \;\&\; d \leq 0 \rightarrow ax^2 + bx + c = e \leftrightarrow d = 0 \;\&\; 2ax + b = 0)$

Преобразуется условие задачи на описание. Выражения x, c и дискриминант d содержат неизвестные. Указатели "группировка(x1)", "группировка(x2)" определяют идентификацию коэффициентов a, b путем группировки всех членов с x^2 и с x. Антецедент обрабатывает дискриминант нормализатором "стандплюс"; если он содержит неизвестные тригонометрические операции, то произведения и степени таких операций преобразуются в суммы. Затем применяется нормализатор "уравнплюс" (приведение подобных членов с неизвестными). Для срабатывания приема необходимо, чтобы ни одно из слагаемых выражения c не делилось на x. Если a, b имеют вхождения неизвестных степеней с дробными показателями, то x должно иметь вхождение тригонометрической операции с неизвестными. Уровень срабатывания равен 5.

Решение простейших степенных неравенств

1. Строгое неравенство с положительным показателем степени.

   $\forall_{abc}(0 < b \,\&\, 0 \le a \to c < a^b \leftrightarrow c < 0 \lor 0 \le c \,\&\, c^{1/b} < a)$

   $\forall_{abc}(0 < b \,\&\, 0 \le a \to a^b < c \leftrightarrow 0 < c \,\&\, a < c^{1/b})$

   $\forall_{abc}(b - \text{rational} \,\&\, \text{числитель}(b) - \text{even} \,\&\, 0 < b \to c < a^b \leftrightarrow c < 0 \lor 0 \le c \,\&\, (a < -c^{1/b} \lor c^{1/b} < a))$

   $\forall_{abc}(b - \text{rational} \,\&\, \text{числитель}(b) - \text{even} \,\&\, 0 < b \to a^b < c \leftrightarrow 0 < c \,\&\, a < c^{1/b} \,\&\, -c^{1/b} < a)$

   $\forall_{abc}(b - \text{rational} \,\&\, \neg(\text{знаменатель}(b) - \text{even}) \,\&\, \neg(\text{числитель}(b) - \text{even}) \,\&\, 0 < b \to c < a^b \leftrightarrow c^{1/b} < a)$

   $\forall_{abc}(b - \text{rational} \,\&\, \neg(\text{знаменатель}(b) - \text{even}) \,\&\, \neg(\text{числитель}(b) - \text{even}) \,\&\, 0 < b \to a^b < c \leftrightarrow a < c^{1/b})$

   Преобразуется неравенство, входящее в условие задачи на описание. Допускаются лишь надутверждения с заголовками "и", "или", "не". Выражение a содержит неизвестные, выражения b, c - не содержат. Уровни срабатывания равны 1 и 3.

2. Строгое неравенство с отрицательным показателем степени. В этом пункте фильтры и указатели приемов - такие же, как в предыдущем пункте.

   $\forall_{abc}(b - \text{rational} \,\&\, \text{числитель}(b) - \text{even} \,\&\, b < 0 \,\&\, \neg(a = 0) \to c < a^b \leftrightarrow c \le 0 \lor 0 < c \,\&\, a < c^{1/b} \,\&\, -c^{1/b} < a)$

   $\forall_{abc}(b - \text{rational} \,\&\, \neg(\text{знаменатель}(b) - \text{even}) \,\&\, \neg(\text{числитель}(b) - \text{even}) \,\&\, b < 0 \,\&\, \neg(a = 0) \to a^b < c \leftrightarrow 0 < ac \,\&\, c^{1/b} < a \lor 0 \le c \,\&\, a < 0)$

   $\forall_{abc}(b - \text{rational} \,\&\, \neg(\text{знаменатель}(b) - \text{even}) \,\&\, \neg(\text{числитель}(b) - \text{even}) \,\&\, b < 0 \,\&\, \neg(a = 0) \to c < a^b \leftrightarrow 0 < ac \,\&\, a < c^{1/b} \lor c \le 0 \,\&\, 0 < a)$

   $\forall_{abc}(b < 0 \,\&\, 0 < a \to c < a^b \leftrightarrow c \le 0 \lor 0 < c \,\&\, a < c^{1/b})$

   $\forall_{abc}(b < 0 \,\&\, 0 < a \to a^b < c \leftrightarrow 0 < c \,\&\, c^{1/b} < a)$

   $\forall_{abc}(b - \text{rational} \,\&\, \text{числитель}(b) - \text{even} \,\&\, b < 0 \,\&\, \neg(a = 0) \to a^b < c \leftrightarrow 0 < c \,\&\, (a < -c^{1/b} \lor c^{1/b} < a))$

3. Строгое неравенство с положительным основанием степени.

   $\forall_{abc}(0 < a \to a^b < c \leftrightarrow 0 < c \,\&\, (0 < b \,\&\, a < c^{1/b} \lor b < 0 \,\&\, c^{1/b} < a) \lor b = 0 \,\&\, 0 < c - 1)$

   $\forall_{abc}(0 < a \to c < a^b \leftrightarrow c \le 0 \lor 0 < c \,\&\, (0 < b \,\&\, c^{1/b} < a \lor b < 0 \,\&\, a < c^{1/b} \lor b = 0 \,\&\, 0 < 1 - c))$

   Приемы применяются к неравенству, входящему в условие задачи на описание. Допустимые внешние символы - "и", "или", "существует". Выражение a содержит неизвестные, выражения b, c - не содержат. Уровни срабатывания равны 2 и 4.

4. Нестрогое неравенство с положительным показателем степени. В этом и следующих пунктах фильтры и указатели приемов - такие же, как у их двойников из предыдущих пунктов.

   $\forall_{abc}(0 < b \,\&\, 0 \le a \to c \le a^b \leftrightarrow c < 0 \lor 0 \le c \,\&\, c^{1/b} \le a)$

$\forall_{abc}(0 < b \;\&\; 0 \leq a \rightarrow a^b \leq c \leftrightarrow 0 < c \;\&\; a \leq c^{1/b})$

$\forall_{abc}(b - \text{rational} \;\&\; \text{числитель}(b) - \text{even} \;\&\; 0 < b \rightarrow c \leq a^b \leftrightarrow c < 0 \;\vee\; 0 \leq c \;\&\; (a \leq -c^{1/b} \;\vee\; c^{1/b} \leq a))$

$\forall_{abc}(b - \text{rational} \;\&\; \text{числитель}(b) - \text{even} \;\&\; 0 < b \rightarrow a^b \leq c \leftrightarrow 0 \leq c \;\&\; a \leq c^{1/b} \;\&\; -c^{1/b} \leq a)$

$\forall_{abc}(b - \text{rational} \;\&\; \neg(\text{знаменатель}(b) - \text{even}) \;\&\; \neg(\text{числитель}(b) - \text{even}) \;\&\; 0 < b \rightarrow c \leq a^b \leftrightarrow c^{1/b} \leq a)$

$\forall_{abc}(b - \text{rational} \;\&\; \neg(\text{знаменатель}(b) - \text{even}) \;\&\; \neg(\text{числитель}(b) - \text{even}) \;\&\; 0 < b \rightarrow a^b \leq c \leftrightarrow a \leq c^{1/b})$

5. Нестрогое неравенство с отрицательным показателем степени.

   $\forall_{abc}(b - \text{rational} \;\&\; \text{числитель}(b) - \text{even} \;\&\; b < 0 \;\&\; \neg(a = 0) \rightarrow c \leq a^b \leftrightarrow c \leq 0 \;\vee\; 0 < c \;\&\; a \leq c^{1/b} \;\&\; -c^{1/b} \leq a)$

   $\forall_{abc}(b - \text{rational} \;\&\; \neg(\text{знаменатель}(b) - \text{even}) \;\&\; \neg(\text{числитель}(b) - \text{even}) \;\&\; b < 0 \;\&\; \neg(a = 0) \rightarrow a^b \leq c \leftrightarrow 0 < ac \;\&\; c^{1/b} \leq a \;\vee\; 0 \leq c \;\&\; a < 0)$

   $\forall_{abc}(b - \text{rational} \;\&\; \neg(\text{знаменатель}(b) - \text{even}) \;\&\; \neg(\text{числитель}(b) - \text{even}) \;\&\; b < 0 \;\&\; \neg(a = 0) \rightarrow c \leq a^b \leftrightarrow 0 < ac \;\&\; a \leq c^{1/b} \;\vee\; c \leq 0 \;\&\; 0 < a)$

   $\forall_{abc}(b < 0 \;\&\; 0 < a \rightarrow c \leq a^b \leftrightarrow c \leq 0 \;\vee\; 0 < c \;\&\; a \leq c^{1/b})$

   $\forall_{abc}(b < 0 \;\&\; 0 < a \rightarrow a^b \leq c \leftrightarrow 0 < c \;\&\; c^{1/b} \leq a)$

   $\forall_{abc}(b - \text{rational} \;\&\; \text{числитель}(b) - \text{even} \;\&\; b < 0 \;\&\; \neg(a = 0) \rightarrow a^b \leq c \leftrightarrow 0 < c \;\&\; (a \leq -c^{1/b} \;\vee\; c^{1/b} \leq a))$

6. Нестрогое неравенство с положителным основанием степени.

   $\forall_{abc}(0 < a \rightarrow a^b \leq c \leftrightarrow 0 < c \;\&\; (0 < b \;\&\; a \leq c^{1/b} \;\vee\; b < 0 \;\&\; c^{1/b} \leq a) \;\vee\; b = 0 \;\&\; 0 \leq c - 1)$

   $\forall_{abc}(0 < a \rightarrow c \leq a^b \leftrightarrow c \leq 0 \;\vee\; 0 < c \;\&\; (0 < b \;\&\; c^{1/b} \leq a \;\vee\; b < 0 \;\&\; a \leq c^{1/b} \;\vee\; b = 0 \;\&\; 0 \leq 1 - c))$

Активизация процедуры перехода к новым неизвестным для показательных неравенств

Если появляется показательное неравенство, содержащее одновременно подвыражения a^{b-c+d} и a^{c-b+f}, где b, c - неизвестны, d, f - известны, то вероятной становится возможность перехода к новой неизвестной a^{b-c}. Для ее проверки отбрасываются блокирующие комментарии (новыенеизвестные ...), введенные при предшествующей преобразованиям попытке перехода к новым неизвестным. Теорема приема имеет вид:

$\forall_{abcdef}(e = c - b + f \rightarrow \text{контекст}(a^{b-c+d}))$

Заголовок приема - "замечание". Он применяется к условию задачи на описание, имеющему заголовок "меньше"либо "меньшеилиравно". Консеквент "контекст(a^{b-c+d})" означает, что попытка применить прием начинается с усмотрения выражения a^{b-c+d}. Указатель "контекст(позиция(x7 корень)вид(x7 степень(x1 x5)))" определяет идентификацию в том же условии выражения a^e, причем первый антецедент, выделенный указателем "идентификатор", проверяет, что e имеет вид $c - b + f$. Указатели "перечень(x6 известно(x6))", "перечень(x4 известно(x4))" определяют идентификацию

f, d как сумм всех известных слагаемых показателей степени. Прием удаляет комментарий задачи (новыенеизвестные меньше) и комментарий текущего условия (новыенеизвестные 0), изменяя вес условия на 0. Таким образом форсируется обращение к процедуре усмотрения новых неизвестных.

Нормализатор общей стандартизации "нормстепень"

Большинство приемов нормализатора представляют собой копии рассмотренных выше приемов упрощения степенных выражений, применяемых при сканировании задачи. Поэтому ограничимся лишь названиями приемов и краткими пояснениями:

1. Возведение десятичного числа в натуральную степень. Если нормализатор имеет комментарий "разложмн", то ограничения на величину показателя степени снимаются.

2. Единица в основании степени.

3. Ноль в основании степени.

4. Единица в показателе степени.

5. Ноль в показателе степени.

6. Повторное возведение в степень.

7. Минус в основании степени.

8. Минус в показателе степени. Исключение минуса в показателе происходит почти без ограничений, кроме некоторых ситуаций, связанных с вычислением пределов.

9. Произведение в основании степени. Преобразование в произведение степеней происходит, если удается усмотреть одинаковые знаки сомножителей, либо при рациональном показателе, имеющем нечетный знаменатель. Единственным средством блокировки является комментарий "свертка".

10. Дробь в основании степени. Приемы аналогичны приемам сканирования задачи, однако фильтры практически отсутствуют - переход к отношению либо произведению степеней выполняется, как только усматривается такая возможность.

11. Устранение модуля в множителях числителя либо знаменателя основания степени.

    $\forall_{abcd}(\neg(a = 0)\ \&\ a - \text{число}\ \&\ \text{числитель}(c) - \text{even}\ \&\ c - \text{rational} \to (d|b|/a)^c = (bd/a)^c)$

    $\forall_{abcd}(\neg(d = 0)\ \&\ d - \text{число}\ \&\ b - \text{число}\ \&\ \neg(b = 0)\ \&\ \text{числитель}(c) - \text{even}\ \&\ c - \text{rational} \to (a/d|b|)^c = (a/bd)^c)$

12. Степень модуля произведения.

    $\forall_{abc}(|ab|^c = |a|^c|b|^c)$

13. Преобразование дробного слагаемого показателя степени к виду суммы дробей. Аналогично описанному выше приему сканирования задачи.

14. Устранение логарифма в показателе степени.

    $\forall_{abcde}(0 < a \ \& \ \neg(a-1=0) \ \& \ 0 < b \to a^{c+d\log_a b/e} = b^{d/e}a^c)$

    $\forall_{abcde}(0 < a \ \& \ \neg(a-1=0) \ \& \ 0 < b \ \& \ \neg(b-1=0) \to a^{c+d/e\log_b a} = b^{d/e}a^c)$

15. Переход от десятичной дроби в основании степени к простой. Если основанием степени служит дробная (содержащая запятую) десятичная константа, а показателем - простая дробь, то основание преобразуется к виду простой дроби.

16. Минус единица в основании степени. Применяются лишь три первых приема из списка аналогичных приемов, срабатывающих при сканировании задачи.

17. Степень сигнума.

    $\forall_{ab}(a - \mathrm{rational} \ \& \ \text{числитель}(a) - \mathrm{even} \to (\mathrm{sg}b)^a = 1)$

    $\forall_{ab}(a - \mathrm{rational} \ \& \ \neg(\text{знаменатель}(a) - \mathrm{even}) \ \& \ \neg(\text{числитель}(a) - \mathrm{even}) \to (\mathrm{sg}b)^a = \mathrm{sg}b)$

    $\forall_{abc}(\neg(\text{знаменатель}(c) - \mathrm{even}) \ \& \ \text{числитель}(b) - \mathrm{even} \ \& \ b - \mathrm{rational} \ \& \ c - \mathrm{rational} \to (\mathrm{sg}a)^{b+c} = (\mathrm{sg}a)^c)$

18. Степень "O".

    $\forall_{abfnx}((x \to a\backslash b) \to (O(f(x)))^n = O((f(x))^n))$

    Выражение n идентифицируется с натуральной константой. f - функциональная переменная.

19. Символы бесконечности.

    $\exp(\infty) = \infty$

    $\exp(-\infty) = 0$

    $\forall_a(0 < a \to \infty^a = \infty)$

    $\forall_a(a - \mathrm{rational} \ \& \ \text{числитель}(a) - \mathrm{even} \to (-\infty)^a = \infty)$

20. Символ не определенного значения.

    $\exp(\text{неопред}) = \text{неопред}$

    Прием обычно используется при вычислении пределов.

21. Одновременное изменение знака слагаемых основания степени с четным показателем для лексикографической стандартизации. Хотя бы одно слагаемое должно иметь заголовок "минус".

22. Извлечение из-под радикала множителя, представляющего собой полный квадрат.

    $\forall_{ab}(\sqrt{a^2b} = |a|\sqrt{b})$

    Прием применяется при наличии комментария "сбросмодуля". Результат замены обрабатывается только нормализаторами общей стандартизации.

Нормализатор стандартизации степеней с неизвестными "уравнстепень"

Нормализатор имеет следующие приемы:

1. Группировка под общий показатель степень степеней с известными основаниями.

   $\forall_{abc}(0 \leq a \ \& \ 0 \leq b \rightarrow a^c b^c = (ab)^c)$

   $\forall_{abc}(0 \leq a \ \& \ 0 < b \rightarrow a^c/b^c = (a/b)^c)$

   $\forall_{abc}(c - \text{rational} \ \& \ \neg(\text{знаменатель}(c) - \text{even}) \rightarrow a^c b^c = (ab)^c)$

   Выражения a, b известны, c содержит неизвестные.

   $\forall_{abcdefg}(c-\text{число} \ \& \ g-\text{rational} \ \& \ \neg(\text{знаменатель}(g)-\text{even}) \ \& \ e-\text{rational} \ \& \ \neg(\text{знаменатель}(e)-\text{even}) \ \& \ \neg(c = 0) \ \& \ \neg(\text{знаменатель}(a)-\text{even}) \ \& \ a-\text{rational} \ \& \ \neg(f = 0) \ \& \ f - \text{число} \rightarrow db^{ae}/fc^{ag} = d(b^e/c^g)^a/f)$

   Выражение a содержит неизвестные, b, c, d, e, f, g известны. Выражения d, e, f, g могут принимать вырожденные единичные значения.

2. Обращение к стандартизации основания степени.

   $\forall_{abc}(b - \text{rational} \ \& \ \neg(\text{знаменатель}(b) - \text{even}) \ \& \ c = a \rightarrow a^b = c^b)$

   $\forall_{abc}(0 \leq a \ \& \ c = a \rightarrow a^b = c^b)$

   Это - рекурсивные обращения к обработке основания степени нормализатором "нормуравн".

3. Обращение к стандартизации показателя степени.

   $\forall_{abc}(0 < a \ \& \ c = b \rightarrow a^b = c^b)$

   Аналогично предыдущему. b - не константа.

4. Дробь в основании степени, имеющей известный показатель.

   $\forall_{abc}(0 \leq a \ \& \ 0 < b \rightarrow (a/b)^c = a^c/b^c)$

   Показатель c, а также числитель либо знаменатель дроби не содержат неизвестных. При этом дробь содержит неизвестные.

Нормализатор решения квадратных уравнений "квадруравн"

В некоторых рассмотренных выше приемах использовался нормализатор "квадруравн", обрабатывающий утверждение, имеющее вид квадратного уравнения. Он позволял сразу перейти к явному решению уравнения. В нормализатор включены следующие приемы:

1. Группировка в левой части уравнения всех неизвестных членов уравнения, а в правой - всех известных.

2. Решение квадратного уравнения общего вида.

3. Решение квадратного уравнения с нулевым свободным членом.

4. Перенесение минуса перед левой частью уравнения в правую часть.

5. Решение однородного уравнения второй степени.

10.9 Упражнения по приемам элементарной алгебры, связанным с арифметическими операциями

10.9.1 Понимание компонент описания приема на ГЕНОЛО́Ге

Естественным способом ознакомится с техникой программирования на ГЕНОЛОГе является анализ уже созданных приемов. Чтобы облегчить читателю начало такой работы, предлагаем несколько упражнений. В них нужно рассмотреть (используя справочные средства интерфейса) основные элементы описания приема и объяснить, для чего эти элементы были введены. Можно пропускать элементы, заведомо относящиеся к другим разделам (интегрирование, дифференциальные уравнения и т.п.). При необходимости следует найти отладчиком ГЕНОЛОГа все задачи раздела "Элементарная алгебра", в которых прием применяется, и попытаться использовать для объяснений контексты его срабатываний. Ниже все подразделы указываются относительно раздела "Элементарная алгебра" оглавления базы приемов.

1. Прокомментировать последний прием подраздела "Минус - Равенство нулю разности $B - A$, если $B - A$ есть 0".

2. Прокомментировать прием подраздела "Цифры - Проверочный оператор "усмне0" - Произведение ненулевых сомножителей".

3. Прокомментировать прием подраздела "Умножение - Раскрывание скобок - Попытка раскрывания скобок для упрощения выражения".

4. Прокомментировать последний прием подраздела "Степени - Общая стандартизация выражений - Простейшие свойства степени - Умножение степеней с одинаковыми основаниями".

5. Прокомментировать последний прием подраздела "Умножение - Нормализатор разложения на множители "видумножение" - Тождества для непосредственного разложения на множители - Сумма кубов".

6. Прокомментировать последний прием подраздела "Умножение - Нормализатор разложения на множители "видумножение" - Разложение многочленов путем подбора целочисленных коэффициентов - Представление многочлена третьей степени в виде произведения линейного множителя и квадратного трехчлена".

7. Прокомментировать прием подраздела "Плюс - Решение уравнений - Системы уравнений - Вычитание линейных уравнений для устранения неизвестной".

8. Прокомментировать последний прием подраздела "Дробь - Устранение квадратичной иррациональности в знаменателе (общий случай)".

9. Прокомментировать прием подраздела "Степени - Решение уравнений - Разбор случаев по знаку неизвестного знаменателя дроби, являющейся основанием степени".

10. Прокомментировать прием подраздела "Число - Нормализатор выделения повторяющихся вхождений числовых выражений "повторчисло" - Умножение - Вынесение за скобку общего неизвестного множителя двух неизвестных слагаемых".

10.9.2 Указания

1. Теорема приема имеет вид $\forall_{ab}(a - b = 0 \to b - a = 0)$. Прием имеет заголовок "второйтерм", и можно предположить, что он выполняет замену $b-a$ на 0. Так как отсутствует указатель, уточняющий способ идентификации антецедента, то этот антецедент должен явным образом присутствовать в контексте заменяемого выражения. Фильтр "уровень(0 3)" определяет попытки применения приема на уровнях 0 и 3. Если на уровне 0 контекст еще не имел утверждения $a-b = 0$, то на уровне 3 такое утверждение может появиться, и прием сработает при повторной попытке.

   В указателях приема находим терм "нормзнака(x2 минус плюс)". Чтобы определить его назначение, выделяем данный терм левой кнопкой мыши, а затем нажимаем правую кнопку. Появляются подсказки "нормзнака(x1 x2 x3)" и "Терм x2(x1) идентифицируется среди операндов операции x3 как множество x2-отрицаний x3-операндов терма, идентифицированного с x1". Следовательно, в нашем случае терм $-b$ идентифицируется среди слагаемых левой части антецедента как сумма взятых с обратными знаками слагаемых выражения b. Хотя лишь одно из имеющих знак "минус" слагаемых заменяемого терма будет идентифицировано с $-a$, рассматриваемый указатель позволит идентифицировать левую часть антецедента с суммой имеющих обратные знаки всех слагаемых заменяемого терма, вне зависимости от числа слагаемых с минусом. Для удаления подсказок дважды нажимаем клавишу "пробел".

   Далее идет указатель "эквивалентно". Чтобы определить его назначение, вызываем на экран подсказку "Теорема, имеющая вид равенства, используется для эквивалентной замены этого равенства на логическую константу "истина" ". Теперь видно, что наше исходное предположение о действии приема ошибочно - на самом деле не $b - a$ будет заменяться на 0, а равенство $b - a = 0$ будет заменяться на константу "истина". Впрочем, пролистывая клавишами "курсор вверх-вниз" другие приемы того же подраздела, нетрудно найти и версию данного приема, лишенную указателя "эквивалентно", т.е. заменяющую $b - a$ на 0. Она срабатывает на уровнях 1 и 3. Рекомендуется самостоятельно, используя отладчик ГЕНОЛОГа, проанализировать возможность удаления одной из данных версий.

2. Теорема приема имеет вид $\forall_{ab}(\neg(a = 0) \,\&\, \neg(b = 0) \to \neg(ab = 0))$. Согласно заголовку "спуск(усмне0)", прием относится к проверочному оператору "усмне0", усматривающему из контекста истинность утверждений вида $\neg(x = 0)$. Текущий анализируемый терм оператора (значение программной переменной x1) - x. Фильтр "уровень(1)" определяет срабатывания приема на уровне 1. Указатель "блокпроверок(1 2)" означает, что антецеденты должны обрабатываться проверочными операторами. Вызываем подсказку для указателя "дистрибразвертка(фикс(0 1 1))". Из нее видно, что фактически прием не группирует сомножители анализируемого выражения в виде произведения двух частей a, b, а по отдельности рассматривает каждый сомножитель и обращается к проверочному оператору для определения отличия этого сомножителя от нуля. Наконец, указатель "спуск" означает, что при неудачной попытке усмотреть отличие от нуля какого-либо сомножителя сразу будет выдан отказ на усмотрение ненулевого значения всего произведения.

3. Теорема приема имеет вид $\forall_{abc}(c = a + b \to a + b = c)$. Так как она выглядит

тавтологично, антецедент должен обращаться к каким-то нормализаторам, выполняющим необходимые преобразования. Наиболее простой способ просмотра нормализаторов - нажатие клавиши "5". В нашем случае после нажатия правая часть антецедента $a+b$ выделяется красным цветом, а под четвертым окном появляется текст нормализатора "стандплюс(замечание(буфер 0))". Таким образом, прием обращается к нормализатору "стандплюс" для обработки встретившейся при сканировании суммы $a+b$. Этому нормализатору передается комментарий (буфер 0), который заблокирует передачу в буферы нормализаторов всех промежуточных результатов раскрывания скобок. Фильтры "уровень(3), "условие", "тип(преобразовать)", "цель(упростить)" определяют срабатывание приема на уровне 3 при рассмотрении условия задачи на преобразование, имеющей цель "упростить". Фильтр "короче(результат теквхожд)" проверяет, что после раскрывания скобок получилось выражение, более короткое, чем исходное.

Далее идет громоздкий фильтр "контекст(вид(теквхожд плюс(умножение(степень(плюс(x4 x5) x6) x7) x8)) не(контекст(подчинено(теквхожд x9)символ(x9 класс) пересекаются(связприставка(x9)x7)не(пересекаются(связприставка(x9) x4)) не(пересекаются(связприставка(x9)x5)))) единица(1 x6 x7)натуральное(x6) заменазнака(минус x7))". Чтобы прочитать его, используем многоцветную указку: нажимаем левую кнопку мыши на слове "вид(...)". Желтым цветом высвечивается фрагмент "вид(теквхожд плюс(умножение(степень(плюс(x4 x5) x6) x7) x8))". Он означает, что преобразуемое выражение $a+b$ должно быть представимо в виде $(d+e)^f g+h$. Таким образом, есть повод обращаться к попытке раскрывания скобок. Передвигая клавишами курсора выделенное желтое пятно, находим сопровождающие указатели "единица(1 x6 x7)", "заменазнака(минус x7)". Они означают, что f, g могут обращаться в единицу, причем коэффициент g может оказаться отрицательным. Можно было бы уточнить, что f, g не должны равняться единице одновременно, однако это излишне: одновременное обращение их в единицу означает наличие в условии задачи вложенных сумм, которые устраняются приемом общей стандартизации, срабатывающим на меньшем уровне. Смещая желтое пятно на второй операнд фильтра, снова сталкиваемся с громоздкой конструкцией, имеющей вид "не(контекст(...))". Нажимаем "курсор вниз" дважды, чтобы можно было читать элементы вложенного "контекст"а. Первые три операнда "подчинено(теквхожд x9)", "символ(x9 класс)", "пересекаются(связприставка(x9)x7)" означают, что рассматриваемая сумма находится под описателем "класс", связывающая приставка которого пересекается со вторым множителем g. Далее идут два операнда, требующие, чтобы эта связывающая приставка не пересекалась с основанием степени $d+e$. Смысл ограничения ясен - здесь сумма $d+e$ играет роль коэффициента, и раскрывание скобок нецелесообразно.

Переходим к следующему фильтру приема - "или(не(контекст(подчинено(теквхожд x4) символ(x4 отображение))) контекст(подчинено(теквхожд x4) символ(x4 сигнум)))". Чтобы лучше понять его смысл, можно было бы провести эксперимент - убрать фильтр, перекомпилировать прием и провести тестовый запуск решателя на нескольких разделах, где вероятно появление описателя "отображение" (например, на математическом анализе). Те задачи, которые при этом перестанут решаться, изменят ответ либо замедлятся, могут прояснить ситуацию. Разумеется, после всего это нужно восстановить прием нажатием клавиши "Shift-Б". Мы не будем здесь прибегать к такому эксперименту, а заметим лишь, что раскрывание скобок при наличии внешнего описателя "отображение" требует особого управления - здесь легко нарушить специфическую стандартизацию, связанную с заданием функций. В порядке исключения, разрешается раскрывать скобки под "сигнумом", который вряд ли окажется связан с подобной стандартизацией.

Фильтр "не(цель(редуцирование))" относится к процедурам базы теорем и нас сейчас не интересует. Наконец, фильтр "или(не(цель(известны))меньше(число(входит(x4 корень))набор(1 2 0)))" блокирует раскрывание скобок в очень громоздких выражениях при упрощении ответа задач, имеющих цель "известно ...". Такие задачи возникают в планиметрии и в аналитической геометрии.

Переходим к указателям приема. Первый из них - "идентификатор(1)" - уточняет способ обработки антецедента. Переменные правой части антецедента будут к моменту его обработки уже идентифицированы, в то время как переменная c еще не определена. Поэтому правая часть обрабатывается нормализатором "стандплюс", и c идентифицируется с результатом обработки. Указатель "попытка(стандплюс теквхожд)" проверяет отсутствие комментария (стандплюс $a+b$). Если его не было, и прием не сработал (из-за того, что новое выражение оказалось сложнее старого), то такой комментарий вводится и блокирует повторные попытки раскрывания скобок. Указатель "замечание(стандплюс результат)" вводит комментарий (стандплюс c), чтобы заблокировать попытки применения приема к выражению c. Этот комментарий будет автоматически корректироваться при последующих преобразованиях c и может пригодиться, если снова появится выражение, в котором станет возможным раскрывание скобок.

4. Теорема приема имеет вид $\forall_{abc}(0 \leq a \rightarrow a^b a^c = a^{b+c})$. Заголовок "второйтерм" определяет замену слева направо. Фильтр "уровень(0 3)" инициирует попытки применения приема на уровнях 0 и 3. Уровень 3 понадобится, если на уровне 0 еще не усматривалась неотрицательность a, но после вывода следствий или эквивалентных преобразований она стала очевидной. Фильтр "или(не(целое(x2))не(десчисло(x1))не(констдробь(x3)))" блокирует срабатывание, если b - целое, a - десятичная константа, c - дробная константа. Например, будет заблокировано нарушающее принятую стандартизацию преобразование $2\sqrt{2}$ в $2^{3/2}$/. Ту же роль играет двойственный фильтр "или(не(целое(x3))не(десчисло(x1))не(констдробь(x2)))".

   Далее идет фильтр "или(не(тип(преобразовать)) не(контекст(комментарий(нормИнтеграл x4)входит(x4 x1)или(контекст(вид(x2 дробь(1 x5)) целое(x3)) контекст(вид(x3 дробь(1 x5))целое(x2))))))". Он означает следующее. Пусть решается задача на преобразование, имеющая комментарий (нормИнтеграл x4), т.е. преобразуется подынтегральное выражение при вычислении неопределенного интеграла, и x4 - переменная интегрирования. Тогда, если x4 входит в a, запрещается, чтобы один из множителей был целочисленной степенью, а другой - радикалом.

   Указатель "блокпроверок(1)" определяет обработку антецедента с помощью проверочного оператора. Указатель "нормализатор" включает процедуру, предпринимающую попытку немедленной общей стандартизации надтермов заменяемого терма после проведения замены. Указатель "единица(1 x2 x3)" разрешает вырожденные значения 1 показателей степени b, c.

5. Теорема приема имеет вид $\forall_{ab}(a^3+b^3=(a+b)(-ab+a^2+b^2))$. Заголовок "замена(второйтерм видумножение)" показывает, что прием относится к нормализатору "видумножение" и выполняет замену слева направо. Фильтр "постпозиция(фикс(0 1 2)фикс(0 1 1))" требует, чтобы второе слагаемое a^3 располагалось в сумме после первого. Таким образом отбрасываются симметричные случаи, и трудоемкость попыток идентификации уменьшается вдвое.

   Далее идет фильтр "коммент(числоценка)", означающий, что при наличии комментария "числоценка" прием блокируется. В принципе, мы могли бы и не уточнять смысл этого ограничения - ясно, что данный комментарий нужен для каких-то специальных случаев. Однако, в качестве иллюстрации попробуем провести небольшое исследование и получить нужную информацию. Попытка найти что-либо полезное в справочнике символа "числоценка" успеха не приносит. Поэтому, чтобы понять смысл данного фильтра, попробуем найти те приемы, которые вводят комментарий "числоценка" при обращении к нормализатору. Возвращаемся в главное меню системы и переходим через него в программу символа "текприем". Это - процедура поиска нужного приема в базе приемов. При сканировании базы приемов на каждом приеме будет происходить обращение к данной процедуре. Нажимаем F3 для получения справки о ее программных переменных. Находим, что значением переменной x5 будет описание текущего приема (объединение списка указателей, нормализаторов и заключенных в общие "скобки" терма "условие(и(...))" фильтров). Возвращаемся в просмотр программы и нажимаем "р" для ее редактирования. После оператора "метка(икс(одиннадцать))" начинаем составление списка условий на отбираемые приемы. Оператор "входит(x11 x5)" позволит нам просматривать все элементы x11 описания приема. Оператор "не(логсимвол(x11))" отбросит те из них, которые являются логическими символами. Таким образом, далее x11 - терм. Нам нужно найти обращения к нормализаторам, содержащие логический символ "числоценка". Поэтому помещаем в программу операторы "входит(числоценка x11)" и "входит(быстрпреобр x11)". По-видимому, пока этим можно ограничиться, и далее помещаем заключительный оператор "ответ(набор(См))". Напомним, что для просмотра будут выбираться те приемы, на которых операторное выражение "текприем" имеет своим значением однобуквенный терм "См". Завершив редактирование программы, возвращаемся в оглавление базы приемов (не входя в просмотр приема) и нажимаем F8. Начнется просмотр всех приемов, удовлетворяющих введенным ограничениям. Переход к очередному приему будет обеспечиваться нажатием клавиши "курсор влево". Заметим, что в цикле поиска невозможно выйти из просмотра приема в оглавление приемов. Если это все же нужно сделать, обрываем цикл просмотра нажатием клавиши "Esc". Тогда возвращение в базу приемов выведет нас на последний просматривавшийся прием. В нашем случае первый найденный прием относится к нормализатору "стандменьшеилиравно". Нажимая клавишу "5", входим в цикл просмотра нормализаторов приема (PageUp-PageDn, выход - End) и обнаруживаем нормализатор "видумножение(замечание(нормуравнение) замечание(числоценка))", который и вводил комментарий "числоценка". Выходя в оглавление, выясняем, что нормализатор "стандменьшеилиравно" используется при стандартизации нестрогих неравенств для известных параметров на этапе редактирования ответа задачи на описание. Видимо, здесь комментарии "нормуравнение" и "числоценка" играли роль опций, ослабляющих попытки разложения на множители в ситуации завершающего редактирования. Очевидно, удлиняющее запись разложение куба суммы на множители здесь нецелесообразно. Снова запускаем цикл просмотра приемов (F8) для поиска других ситуаций использования комментария. Пропуская несколь приемов, аналогичных рассмотренному, в некоторый момент обнаруживаем прием для разложения на множители числителя: $\forall_{abcdef}(f = b{+}c \,\&\, d{-}\mathrm{rational} \,\&\, \neg(\text{знаменатель}(d){-}\mathrm{even}) \to a(b{+}c)^d/e = af^d/e)$. Так как его описание достаточно громоздкое, нажимаем "Ctr-F3" и убираем с экрана третье окно. Затем нажимаем "5" и просматриваем нормализаторы. Обнаруживаем громоздкий нормализатор "видумножение(...)", внутри которого усматривается фрагмент "замечание(условие(и(константа(x2)константа(x3)десчисло (x5))) числоценка)". Он означает, что комментарий "числоценка" будет вводиться, если выражения b, c, e - константные. Видимо, изначальное происхождение данного комментария - ориентировать нормализатор на работу с константными выражениями. Некоторые действия (типа преобразования суммы кубов) здесь будут блокироваться, некоторые (типа вынесения наружу общего натурального множителя целочисленных коэффициентов суммы) - активироваться. Судя по первому приему, позднее этот комментарий был использован в прочих ситуациях как дополнительный способ ограничения разложения на множители.

Возвращаемся к рассмотрению компонент описания приема для суммы кубов. Следующий фильтр - "или(входит(нормИнтеграл комментарии)не(контекст(комментарий(нормИнтеграл x3)тригаргумент(корень x4)входит(x3 x4))))". Если обращение к разложению на множители произошло из задачи преобразования подынтегрального выражения, причем переменная интегрирования встречается под тригонометрической операцией, то прием применяется только при наличии комментария "нормИнтеграл". Такие комментарии вводятся нормализатором неопределенного интегрирования "нормИнтеграл" при некоторых рекурсивных обращениях.

Еще один фильтр из математического анализа - "не(контекст(текущаязадача описать)цель(текущаязадача областьграницы))". Прием блокируется, если решается задача на явное задание области по системе ее граничных линий. Рекомендуется в качестве упражнения найти ту задачу из раздела "Кратные интегралы", на решение которой повлияет отмена данного фильтра. Для этого фильтр удаляется, прием перекомпилируется, и выполняется цикл решения задач в указанном разделе.

Фильтр "не(контекст(тип(текущаязадача преобразовать)или(цель (текущаязадача смпосылка) цель(текущаязадача новаяпеременная)цель(текущаязадача значениеинтеграла)цель(текущаязадача редакция))))" перечисляет целевые установки задач на преобразование, при которых выполнение приема нецелесообразно. Как и в случае предыдущего фильтра, рекомендуем найти задачи, для которых отсутствие данного ограничения приводит к трудностям - замедлению либо утере ответа. Так как многие фильтры вводились при обучении ради единственного примера, повторный анализ, по накоплении многих примеров, может привести к существенному усовершенствованию решающих правил. Наличие циклов тестирования по задачнику обычно позволяет модифицировать даже очень старые приемы, сохраняя полную работоспособность решателя на освоенном обучающем материале. Если коррекция приема приводит к утере ответов и замедлению решений, то обычно достаточно бывает создать дополнительные приемы, реже - приходится продолжать коррекцию других приемов. Разумеется, могут встретиться особые точки, для которых процесс коррекций и модификаций ветвится и требует очень большой работы, однако чаще всего глубина модификаций невелика. Размеры решателя не должны пугать тех, кто столкнется с "плохими" приемами и будет вынужден их переделывать - опыт показывает, что такие переделки, как правило, имеют локальный характер.

Фильтр "не(контекст(тип(текущаязадача описать)цель(текущаязадача неравенства)))" блокирует применение приема, если решается задача на описание для определения области интегрирования.

Указатель "уровень(4)" определяет срабатывание приема нормализатора на уровне 4. Указатель "знаксуммы(минус фикс(0 1))" определяет попытку идентификации с одновременным изменением знаков кубов на "минус". При этом будет изменен знак заменяющего терма. Указатель "модификатор" требует, чтобы сумма не имела других слагаемых. Указатель "замечание(6 фикс(0 2 2))" вводит комментарий (6, $-ab+a^2+b^2$), который заблокирует попытку разложения второго сомножителя по формуле корней квадратного трехчлена. Рекомендуем найти все использующие этот комментарий приемы нормализатора "видумножение", написав подходящую программу поиска "текприем". Указатель "замечание(условие(и(входит(нормчислитель комментарии) контекст(тригаргумент(корень x4)))) фильтрудвоения)" вводит комментарий "фильтрудвоения", если нормализатор имел комментарий "нормчислитель", а преобразуемое выражение имело тригонометрические операции. Рекомендуем выяснить роль комментария "фильтрудвоения", найдя с помощью процедуры "текприем" все приемы нормализатора, фильтры которых содержат ссылку на этот комментарий.

6. Теорема приема имеет вид $\forall_{abcdefghijk}(fh = d \;\&\; gj = a \;\&\; 0 < g \;\&\; gi = b - fj \;\&\; gh + fi = c \to ae^3 + be^2k + cek^2 + dk^3 = (fk + ge)(hk^2 + iek + je^2))$. Заголовок такой же, как в предыдущем случае - "замена(второйтерм видумножение)". Фильтры "целое(x1)", "целое(x2)", "целое(x3)", "целое(x4)" определяют идентификацию коэффициентов a, b, c, d с целочисленными константами. Фильтр "постпозиция(фикс(0 1 4)фикс(0 1 1))" требует, чтобы dk^3 идентифицировалось со слагаемым, идущим после слагаемого ae^3. Ввиду симметрии теоремы, это вдвое сокращает трудоемкость попыток идентификации. Фильтр "меньше(количествооперандов(теквхожд)5)" - ускоряющий. Он размещается компилятором в начале программы, так как не содержит каких-либо переменных, которые вначале нужно было бы идентифицировать. Поэтому заведомо неприемлемые длинные суммы сразу отбрасываются. Фильтр "коммент(группировка)" блокирует применение приема, если внешняя процедура "видумножение" обратилась к данной процедуре для разложения на множители выражения, в котором часть слагаемых уже сгруппирована. Фильтр "контекст(разряд(теквхожд степень x12)не(второйсимвол(x12 2)))" тоже ускоряющий. Он обрабатывается до начала идентификации слагаемых и отбрасывает суммы, не содержащие степеней, кроме, быть может, квадратов. Фильтр "коммент(6 теквхожд)" блокирует разложение на множители трехчленов, возникших при разложении на множители суммы или разности кубов. Фильтр "не(константа(корень))" отбрасывает константные суммы. Фильтр "коммент(нормуравнение)" блокирует прием, если имеется комментарий "нормуравнение", указывающий на ослабленный режим разложения на множители. Аналогичную роль играет фильтр

"коммент(множителимн)". Наконец, фильтр "меньше(2 количествооперандов)" блокирует попытку идентификации для суммы с двумя слагаемыми.

Указатель "модификатор" запрещает наличие в сумме слагаемых, не указанных в теореме. Указатель "уровень(3)" определяет срабатывание приема нормализатора на уровне 3. Указатель "программа(1 2 3 4 5)" определяет реализацию всех антецедентов с помощью арифметических вычислений. Указатели "подстановка(фикс(0 1 2)x2 0)", "подстановка(фикс(0 1 3)x3 0)" разрешают обращение в 0 коэффициентов b, c. Так как суммы длины 2 не рассматриваются, хотя бы один из этих коэффициентов окажется отличным от 0. Указатели "перечень(x1 десчисло(x1))", "перечень(x4 десчисло(x4))" определяют идентификацию a, d с произведением всех численных коэффициентов соответствующих слагаемых. Лишь после этого будут выполняться попытки усмотреть в остатках их множителей кубы некоторых выражений e, k. Указатель "пересечениесписков(фикс(0 1 2)фикс(0 1 3))" объясняет компилятору, что второе и третье слагаемое могут не иметь своим заголовком символ "умножение", хотя каждое из них в теореме содержит по три сомножителя. Как следствие, компилятор не будет уточнять заголовки этих слагаемых. Корректная идентификация будет обеспечиваться применением процедур, задаваемых справочником "пересечениесписков". Указатель "единица(1 x1 x2 x3 x4)" разрешает обращаться коэффициентам a, b, c, d в единицу. Указатель "заменазнака(минус x1 x2 x3 x4)" определяет передачу этим коэффициентам знака "минус" перед слагаемым. Наконец, указатель "титр(набор(1 терм(фикс(0 2)фикс(1)фикс(2)фикс(4)фикс(5))параметры(x6 x7 x8 x9 x10)))" определяет составление сопровождающих срабатывание приема пояснений. Для просмотра шаблона текста пояснений нажимаем клавишу "6". Под изображением приема появляется запись "Подбираем целочисленные коэффициенты разложения (), используя соотношения (),(),(),() ". Элемент указателя "терм(фикс(0 2)фикс(1)фикс(2)фикс(4)фикс(5)" ссылается на подставляемые вместо пар скобок термы. Элемент "параметры(x6 x7 x8 x9 x10)" уточняет, что вместо вспомогательных переменных f, g, h, i, j в текст пояснения будут подставляться первые неиспользуемые большие буквы.

7. Теорема приема имеет вид: $\forall_{abcdefgx}(agx + b = c \ \& \ \neg(a = 0) \rightarrow dgx + e = f \leftrightarrow ae - bd = af - cd)$.

   Заголовок - "второйтерм", т.е. прием выполняет эквивалентную замену слева направо. Фильтры "уровень(1)", "тип(описать)", "условие", "корень" определяют применение приема к условию задачи на описание на уровне 1. Первый антецедент не выделен указателями и идентифицируется с другим условием. Фильтр "неизвестные(2)" означает, что задача должна иметь более одной неизвестной. Фильтры "неизвестная(x23)", "известно(x1)", "известно(x4)", "известно(x3)", "известно(x6)", "не(входит(x23 x2))", "не(входит(x23 x5))" указывают, что x - неизвестная, выражения a, d, c, f не содержат неизвестных, а выражения b, e не содержат неизвестной x. Таким образом, оба уравнения задачи линейны относительно x. Фильтр "не(контекст(неизвестная(x8)входит(x8 x2) не(входит(x8 x5))))" требует, чтобы каждая неизвестная выражения b входила в выражение e. Тогда новое уравнение не приобретет новых неизвестных, а старая неизвестная x окажется исключенной. Фильтр "или(не(линейноеуравнение(корень неизвестные))линейноеуравнение(фикс(1)неизвестные))" блокирует применение приема, если он нарушает линейность текущего уравнения, комбинируя его с нелинейным. Фильтр "не(Входит(известно цели))" блокирует применение приема в задаче, имеющей цель (известно ...), так как эта задача решается путем вывода следствий в блоке анализа. Фильтр "не(Входит(независит цели))" блокирует применение приема в задаче, имеющей цель (независит ...), запрещающей зависимость ответа от заданных переменных. Такие задачи обычно решаются другими средствами - путем приравнивания нулю неизвестного коэффициента перед теми переменными, которые должны быть устранены.

Указатель "блокпроверок(2)" определяет обработку второго антецедента проверочным оператором. Указатели "единица(0 x2 x5)", "единица(1 x1 x4 x7)" разрешают вырожденные нулевые значения b, e и единичные значения a, d, g. Указатель "заменазнака(минус x1 x4)" обеспечивает передачу коэффициентам a, d знака "минус" перед слагаемым. Указатель "титр(набор(1 терм(x23 фикс(фикс(1))x1 минус(x4))))" определяет подстановку в шаблон поясняющего текста необходимых термов. Указатель "комментарий(2 титр(стоп))" передает проверочному оператору "усмне0", обрабатывающему второй антецедент, комментарий (титр стоп), блокирующий отображение на экране обращения к проверке.

Для просмотра нормализаторов приема нажимаем клавишу "5", и далее сменяем текущий нормализатор клавишами "PageUp-Page-Dn". Обратим внимание на нормализаторы "стандплюс(замечание(титр(набор(2))))", "стандплюс(замечание(титр(набор(3))))", обрабатывающие разности $ae - bd$, $af - cd$ соответственно. Термы "замечание(титр(набор(...)))" обеспечивают выдачу на экран второго и третьего фрагментов сопровождающего текста, комментирующих обращения к этим нормализаторам.

8. Теорема приема имеет вид $\forall_{abcdefgpq}(\neg(a + b = 0)\ \&\ g = a - b\ \&\ p = a^2\ \&\ q = b^2\ \&\ c = p - q\ \&\ (\neg(g = 0)\ \vee\ \neg(c = 0)) \rightarrow f/d(a + b)^e = f(g/c)^e/d)$. Заголовок приема - "второйтерм", т.е. выполняется тождественная замена слева направо. Фильтры "уровень(4)", "условие", "тип(преобразовать)", "цель(упростить)" определяют срабатывание приема на уровне 4 в условии задачи на преобразование, имеющей цель "упростить". Фильтры "не(цель(нормИнтеграл))", "коммент(длина)" блокируют применение приема при интегрировании и на этапе заключительной обработки ответа. Фильтр "контекст(вид(x1 умножение(x9 степень(x10 дробь(x11 умножение(2 x12)))))или(заголовок(x10 плюс)контекст(алгебрвхождение(x10 x13)заголовок(x13 плюс)))заменазнака(минус x9)единица(1 x9 x12))" требует, чтобы выражение a имело своим сомножителем степень с дробным показателем, знаменатель которого делится на 2. При этом требуется, чтобы основание степени либо являлось суммой, либо содержало сумму, достижимую из корня через алгебраические операции. Фильтр "не(контекст(вид(x2 плюс(x9 умножение(x10 степень(x11 дробь(x12 x13)))))не(константа(x11)) заменазнака(минус x10)единица(1 x10)))" означает, что b не должно представлять собой невырожденную сумму, некоторое слагаемое которой имеет своим сомножителем степень с дробным показателем и неконстантным основанием. Следующий фильтр имеет ссылку на переменную c, отсутствующую в заменяемом выражении. Поэтому временно откладываем чтение фильтров и переключаемся на указатели, определяющие обработку антецедентов.

Указатель "идентификатор(2 3 4 5)" означает, что антецеденты со второго по пятый представляют собой равенства, используемые для ввода в рассмотрение новых объектов. Второй антецедент идентифицирует переменную g с разностью выражений a и b. На нее будут домножаться числитель и знаменатель для исключения иррациональности. К разности применяются нормализаторы общей стандартизации. Третий и четвертый антецеденты идентифицируют выражения p, q с квадратами выражений a, b, обработанными нормализаторами "нормстепень" и "стандплюс". Наконец, пятый антецедент идентифицирует переменную c с разностью квадратов, обработанной нормализаторами "нормплюс" и "видумножение".

Теперь возвращаемся к третьему окну описания приема. Следующий фильтр имеет вид "не(контекст(вид(x3 плюс(x9 умножение(x10 степень(x11 дробь(x12 умножение(2 x13))))))единица(1 x10 x13)заменазнака(минус x10)))". Он проверяет, что разность квадратов c не имеет слагемого, сомножителем которого была бы степень с дробным показателем, имеющим четный знаменатель. Это означает, что от иррациональности в знаменателе удалось избавиться. Фильтр "не(константа(фикс(0 1 2 2 1)))" требует, чтобы выражение $a + b$ не было константным - для этого случая созданы отдельные приемы. Фильтр "не(контекст(алгебрвхождение(фикс(0 1 2 2 1)x9)символ(x9 дробь)))" блокирует применение приема, если выражение $a + b$ имеет дробь, достижимую из его корня через алгебраические операции. В таких ситуациях сначала должно быть выполнено исключение дробей (например, путем сложения дробных выражений). Наконец, фильтр "меньше(числочленов(x3 плюс)плюс(числочленов(x15 плюс)числочленов(x16 плюс)))" требует, чтобы при вычитании b^2 из a^2 произошло приведение подобных членов, уменьшившее число слагаемых. Без этого преобразование будет фиктивным - разложение на множители знаменателя и сокращение дроби вернут к исходному выражению.

Возвращаемся к рассмотрению указателей приема. Указатель "блокпроверок(1 6)" определяет обработку первого и шестого антецедентов проверочным оператором. Первый антецедент почти избыточен - он перепроверяет о.д.з. исходного выражения. Однако, явное упоминание его в теореме будет использовано указателем "вывод(не(равно(x3 0))эквивалентно(1))", выводящим дополнительное следствие $c \neq 0$ и регистрирующим его эквивалентность (в предположении истинности прочих антецедентов) первому антецеденту. Шестой антецедент проверяет, что либо $a - b \neq 0$, либо $a^2 - b^2 \neq 0$. Иногда последнее оказывается легче проверить, чем первое. Таким образом обеспечивается корректность преобразования. Указатели "заменазнака(минус x6)" и "единица(1 x4 x5)" разрешают вырожденные единичные значения d, e и отнесение внешнего минуса к f. Наконец, указатель "титр(набор(1 терм(фикс(фикс(0 1))степень(x7 x5))))" определяет выражения, подставляемые в шаблон пояснений к срабатыванию приема.

9. Теорема приема имеет вид $\forall_b(\neg(b = 0) \rightarrow 0 < b \ \vee \ b < 0)$. Указатель "контрольвывода(степень(дробь(x1 x2)x3))" определяет инициализацию приема при усмотрении выражения вида $(a/b)^c$. Заголовок приема - "выводусловия", т.е. происходит занесение утверждения $0 < b \ \vee \ b < 0$ в список условий задачи. Фильтры "уровень(0)", "условие", "тип(описать)" означают, что прием применяется на уровне 0 в задаче на описание, причем усмотренное выражение $(a/b)^c$ входит в условие задачи. Фильтр "не(известно(x2))" требует, чтобы знаменатель b содержал неизвестные. Фильтр "контекст(вид(корень равно(x4 x8))не(известно(x4))или(заголовок(x8 0)контекст(подтерм(умножение(теквхо-

жд x2 x9))единица(1 x9)))не(контекст(вид(x4 плюс(x5 x6))не(контекст(обобщмножитель(x5 x7)или(контекст(вид(x7 x1))контекст(вид(x7 x2))))) единица(0 x6))))" достаточно громоздкий, поэтому читаем его по частям, используя выделение фрагментов многоцветной указкой. Получаем, что текущее условие задачи должно иметь вид уравнения "равно(x4 x8)", причем часть x4 содержит неизвестные, а часть x8 либо нулевая, либо имеет вид $(a/b)^c \cdot b \cdot A$. Кроме того, каждое слагаемое x5 части x4 должно иметь своим обобщенным множителем либо a, либо b. Выполнение этого условия означает, что после разбора случаев, возможно, удастся сократить уравнение на b или на a. Фильтр "не(контекст(условие(x4)входит(разборслучаев комментарииусловия(x4))))" означает отсутствие условия, помеченного комментарием "разборслучаев". Таким образом, сначала должны быть реализованы разборы случаев, инициированные ранее. Фильтр "неизвестные(1)" говорит, что задача должна иметь единственную неизвестную. В противном случае срабатывание приема на уровне 0 может оказаться слишком поспешным. Фильтр "не(контекст(подтерм(равно(умножение(теквхожд x4)0))единица(1 x4)заменазнака(минус x4)))" блокирует срабатывание приема в вырожденной ситуации, когда уравнение имеет вид $(a/b)^c{\cdot}A = 0$ и разбор случаев не нужен. Фильтр "не(входит(связка цели))" блокирует применение приема при решении дифференциальных уравнений. Фильтры "не(легковидеть(меньше(0 x2)))", "не(легковидеть(меньше(x2 0)))" блокируют срабатывание приема в ситуациях, когда знак знаменателя усматривается из контекста.

Указатель "примечание(разборслучаев)" сопровождает выведенную дизъюнкцию комментарием "разборслучаев", форсирующим ускоренный переход к рассмотрению случаев. Кроме того, данный комментарий блокирует возможные попытки усмотреть тавтологичность дизъюнкции и заменить ее на константу "истина". Указатель "блокпроверок(1)" определяет обработку антецедента проверочным оператором. Наконец, указатели "титр(набор(1 терм(x2)))", "комментарий(1 титр(стоп))" обеспечивают сопровождение срабатывания приема пояснениями.

10. Теорема приема имеет вид $\forall_{abc}(ab+ac = a(b+c))$. Фильтры "не(известно(x1))", "не(известно(x2))", "не(известно(x3))", "неизвестные(2)" означают, что как общая часть a двух слагаемых, так и остаточные произведения b, c содержат неизвестные, причем общее число неизвестных задачи более единицы. Фильтр "или(контекст(внешвхождение(x4)вид(x4 плюс(степень(x2 x6)степень(x3 x6) x5))единица(0 x5)единица(1 x6)буфер(новыенеизвестные базавхождения(x4))) и(контекст(вид(x2 минус(x4))вид(x3 минус(x5))контекст(внешвхождение(x6) вид (x6 плюс(степень(x4 x7)степень(x5 x7)x8))единица(0 x8)единица(1 x7) буфер(новыенеизвестные базавхождения(x6))))))" анализирует контекст вхождения текущей суммы. Этот контекст состоит из вхождений, расположенных вне суммы в том же терме задачи, либо в других термах, отобранных для усмотрения повторяющихся вхождений выражений. Первая часть фильтра указывает на существование в контексте вхождения x4 выражения вида $b^k + c^k + r$, вторая - вхождения x6 выражения вида $(-b)^k + (-c)^k + r$. Элементы "буфер(новыенеизвестные базавхождения(x4))", "буфер(новыенеизвестные базавхождения(x6))" нужны для регистрации в комментарии к нормализатору (новыенеизвестные ...) ссылок на те термы задачи, в которых обнаруживаются указанные вхождения. По окончании преобразований текущего терма будет

предпринято повторное сканирование имеющихся в данном комментарии термов.

10.9.3 Анализ целесообразности применения приема

При обучении решателя ранее введенные фильтры приема могут устаревать из-за появления новых приемов или изменения старых. В новой задаче, где возникает срабатывание приема, оно может оказаться бесполезным и заводящим задачу в тупик. Наоборот, иногда старый фильтр приема оказывается чрезмерно жестким и блокирует необходимое для решения новой задачи срабатывание. Во всех этих ситуациях коррекция фильтра, подсказываемая новой задачей, требует учета возможных срабатываний приема в других задачах. Здесь различаются два случая. Если нужно усилить ограничения на срабатывание приема, то достаточно рассмотреть лишь те задачи, в которых прием срабатывал раньше. Их список быстро определяется отладчиком ГЕНОЛОГа. После усиления фильтра выполняется повторная прокрутка найденных задач и анализируются возможные коллизии. Во втором случае нужно ослабить ограничения на срабатывание приема. Это - более трудоемкая работа, так как требует уже полной прокрутки по всем разделам задачника, где прием может сработать. Для иллюстрации использования в таких ситуациях отладчика ГЕНОЛОГа разберем ряд примеров, относящихся к вышеизложенным первым подразделам элементарной алгебры.

1. Найти все задачи раздела "Элементарная алгебра", в которых срабатывает последний прием подраздела "Умножение - Усмотрение разности квадратов - Усмотрение разности квадратов". В каких задачах его срабатывание не нужно?

2. Найти все задачи раздела "Элементарная алгебра", в которых срабатывает последний прием подраздела "Дробь - Устранение кубической иррациональности в знаменателе". Выяснить, к каким последствиям для решения прочих задач данного раздела приведет отбрасывание двух последних фильтров "не(константа(фикс(0 1 2 2 1)))", "меньше(числочленов(x3 плюс) плюс(числочленов(x15 плюс)числочленов(x16 плюс)))".

3. Найти все задачи раздела "Элементарная алгебра - Упрощение выражений", в которых срабатывает последний прием подраздела "Степени - Разбор случаев по знаку сомножителя под радикалом при упрощении выражений". Выяснить, к каким последствиям приведет отбрасывание фильтра "контекст(подчинено(теквхожд x4)символ(x4 степень)второйсимвол(x4 дробь)второйсимвол(второйоперанд(x4)2 4))".

4. Найти задачи раздела "Элементарная алгебра", в которых последний прием подраздела "Степени - Решение уравнений - Показательные уравнения - Представление произведения неизвестных степеней с известными основаниями в виде степени произведения этих оснований, домноженной на известный коэффициент", - имеет ненужное срабатывание. Найти все задачи того же раздела, которые перестают решаться при отключении данного приема. Попробовать усилить фильтры таким образом, чтобы прием срабатывал лишь в последних задачах.

5. Найти задачи раздела "Элементарная алгебра", в которых прием подраздела "Степени - Решение уравнений - Выражение одной неизвестной через другую,

если известно произведение их степеней" - имеет ненужное срабатывание. Решение каких задач данного раздела задачника нарушится, если у приема отбросить фильтр "конец(альтернатива(уровень(3)...))" ?

10.9.4 Указания

1. Начинаем с того, что заходим в корневое меню оглавления задачника и выделяем пункт "Элементарная алгебра". Затем нажимаем клавишу "Shift-2" и попадаем в оглавление приемов. Переходим в просмотр нужного нам приема и нажимаем клавишу "А" (кириллица). После небольшой паузы, необходимой программе для поиска по разделу задачника, нажимаем "Shift-3" и возвращаемся в оглавление задачника. Все найденные задачи, имеющие хотя бы одно срабатывание рассматриваемого приема, оказались зарегистрированы в буфере задачника. Чтобы перейти к просмотру буфера, нажимаем "ф". В буфере обнаруживаем единственный подраздел "Усмотрение разности квадратов", название которого копирует название соответствующего концевого пункта оглавления приемов. Входим в данный подраздел. В нем оказывается список из двух десятков номеров отобранных задач. Здесь дублируются только ссылки на задачи, но не сами задачи. Поэтому, изменяя задачу через буфер, одновременно изменяем ее и по "основному" месту регистрации в задачнике. Однако, удаление задачи из буфера не означает удаления основной ссылки на нее. Для анализа срабатываний приема на отобранных задачах можно либо включить цикл "прокрутки", начинающейся с текущей выделенной задачи (клавиша "Ctr-з"), либо предпринять трассировку отдельных задач. Чтобы определить, в каких задачах срабатывание приема необходимо, выделяем первый пункт списка отобранных задач и нажимаем клавишу "й". Это вызовет цикл попыток решения данных задач с заблокированным приемом. В тех случаях, когда ответ не изменяется и находится быстрее, чем при использовании приема, пункт списка помечается знаком вопроса. Для нашего примера таких знаков вопроса оказывается два - на пунктах 16 и 17. Можем посмотреть, например, срабатывание приема в пункте 16. Для этого заходим в просмотр задачи и нажимаем "г" для перехода в оглавление приемов. Далее выходим в просмотр нашего приема и нажимаем "Ctr-Enter". При попытке обратиться к приему происходит прерывание отладчика ЛОСа. На экране видна лишь часть выполняемого фрагмента программы, обрывающаяся на текущем операторе. Для просмотра всего фрагмента нажимаем "ф". Нас интересует заключительный момент срабатывания приема, поэтому далее ведем трассировку через отладчик ЛОСа. Нажимаем клавишу "курсор вниз" для выделения оператора программы и перемещаем выделенный оператор клавишей "курсор вправо", пока он не достигнет оператора "ветвь 1". Вместо использования клавиш курсора можно просто подвести курсор мыши к указанному оператору и нажать левую кнопку (отказ от выделения оператора - путем нажатия клавиши "курсор вверх"). Снова нажимаем "Ctr-Enter", чтобы отладчик ЛОСа выполнил действия вплоть до выделенного оператора включительно. Для прорисовки полного фрагмента программы опять нажимаем "ф". Видно, что сейчас будет выполняться реализующий действия приема оператор "заменавхождения". Чтобы просмотреть его выполнение на уровне трассировки ГЕНОЛОГа, нажимаем клавиши "пробел" и "Enter". Появляется кадр "Условие $(a-b)(b-a)$ заменяется на $-a^2+b^2$". Можно просмотреть описание приема, нажав клавишу "б". Если также нажать "п", то на экране

появится группа равенств, определяющих для каждой теоремной переменной то значение, которое она получила при данном срабатывании. Через отладчик ЛОСа нетрудно выяснить, что прием был использован при упрощении условия $\neg((a+b)(a-b)=0)$. Очевидно, здесь его срабатывание действительно ненужно. По завершении упражнения сбрасываем буфер задачника, нажимая из оглавления клавишу "О" (кир.)

2. Так же, как в предыдущем случае, создаем в буфере задачника список задач, имеющих срабатывание рассматриваемого приема. Оказывается, что в нем всего одна задача - на упрощение выражения. Затем возвращаемся в просмотр приема, входим в редактирование третьего окна (Ctr-3) и удаляем два последних фильтра. По окончании редактирования нажимаем "Enter" и "F3". Таким образом возникает измененная версия приема. Переходим в оглавление задачника, входим в меню подраздела "Элементарная алгебра" и выделяем первый пункт этого меню. Для прокрутки решателя по всему подразделу нажимаем "Ctr-з" (кир.). Начинается цикл решения задач. После него снова оказываемся в меню подраздела "Элементарная алгебра". Для просмотра результатов прокрутки нажимаем клавишу "з". Появляется диалоговый блок со статистикой прокрутки, из которого видно, что ни одна задача не замедлилась и не изменила ответа. Это означает, что данные фильтры, возможно, вообще не нужны. Они были введены из общих соображений, по аналогии с приемами для устранения квадратичной иррациональности. При желании их можно удалить, не исказив поведения решателя на обучающем материале. В качестве упражнения рекомендуем придумать задачу, которая решалась бы по-разному при наличии этих фильтров и при их удалении. По окончании упражнения возвращаемся в просмотр измененного приема и восстанавливаем его первоначальный вид нажатием клавиши "Б".

3. Находим список всех задач, имеющих срабатывание приема, используя указанный выше интерфейс отладчика ГЕНОЛОГа. Этот список содержит две задачи. Отбрасываем указанный в упражнении фильтр, перекомпилируем прием нажатием F3 и запускаем прокрутку решателя по рассматриваемому подразделу задачника. Анализируя результаты прокрутки, обнаруживаем, что 4 задачи изменили свои ответы, одна задача замедлилась более чем на 4 млн. шагов интерпретатора ЛОСа и три задачи - примерно на 1 млн. Суммарное замедление по разделу составило 9 млн. шагов. Чтобы подробнее рассмотреть причины изменений, нажимаем из диалогового блока просмотра статистики прокрутки клавишу "и". Попадаем на первую задачу с изменившимся ответом. Чтобы пропустить текущую задачу и перейти к следующей задаче списка задач с измененным ответом, будем каждый раз нажимать клавишу "ш". Однако, нужно помнить, что запуск решения любой задачи задачника сбрасывает буфер, хранящий список отобранных для просмотра задач. Тогда дальнейшие нажатия "ш" ничего не дадут, и для возобновления просмотра списка с начала нужно будет снова нажать "и" из диалогового блока.

   Продолжаем анализ первой задачи списка измененных ответов. В ней нужно было разложить на множители $a\sqrt{ba}-\sqrt{ba}-ba+a$. Исходный ответ имел вид $(-\sqrt{ab}+a)(\sqrt{ab}+1)$. Видно, что новый ответ более громоздкий и состоит из двух случаев: при $a \leq 0, b \leq 0$ имеем выражение $(-\sqrt{-a}\sqrt{-b}+a)(\sqrt{-a}\sqrt{-b}+1)$, иначе - $(\sqrt{a}\sqrt{b}+1)(-\sqrt{b}+\sqrt{a})\sqrt{a})$. Разбор случаев здесь позволил измельчить выражения под радикалами, однако неясно, насколько полезно такое измельчение. Чтобы выяснить, что давал отброшенный фильтр, восстанавливаем программу приема нажатием "Б", и запускаем найденные выше две задачи, в которых срабатывала его исходная версия. Выясняется, что в обоих случаях применение приема позволяло усмотреть полный квадрат в сумме под радикалом. Это и объясняет появление фильтра.

4. Список задач раздела "Элементарная алгебра", в которых срабатывает рассматриваемый прием, имеет четыре элемента. Чтобы найти те из них, где прием не нужен, выделяем первый элемент списка и нажимаем "й". Знак вопроса появляется около третьей задачи. В ней требуется решить уравнение $(15\sqrt{3}+26)^x - 5(4\sqrt{3}+7)^x + 6(\sqrt{3}+2)^x + (-\sqrt{3}+2)^x = 5$. Чтобы найти задачи, которые перестают решаться при отключении приема, входим в просмотр его описания и нажимаем F7. Это уничтожает программу приема. При запуске решателя на отобранной в буфере четверке задач выясняется, что два ответа утеряны. Это первая и вторая задачи. В них требовалось решить уравнения $10^x - 5^{x-1}2^{x-2} = 950$, $a^{x+2}b^{x+3} = 4(ab)^{2x} + 1$. Последняя задача списка, хотя и замедляется немного (примерно на 20 процентов), все же остается решенной. Ее условие - неравенство $6 \cdot 9^{1/x} - 13 \cdot 3^{1/x} \cdot 2^{1/x} + 6 \cdot 4^{1/x} \leq 0$. Нам нужно найти какой-то простой признак, отличающий срабатывания приема в первых двух задачах от срабатываний в двух последних. Для этого переходим в просмотр приема, восстанавливаем его программу нажатием клавиши F5 и возвращаемся в буфер задачника. Используя отладчик ЛОСа, анализируем точки срабатывания приема в указанных задачах. При этом выясняется, что в первых двух случаях условие уже имело степень вида $(ad)^X$ с неизвестным показателем X, а в последних двух - не имело. Поэтому создаем дополнительный фильтр "контекст(позиция(x6 корень)вид(x6 степень(x7 x8))не(известно(x8))равно(x7 терм(умножение(x1 x4))))". Он проверяет наличие в том же условии степени "степень(x7 x8)" с неизвестным показателем x8 и основанием, равным результату обработки нормализатором "нормумножение" выражения ad.

5. Создаем в буфере задачника список всех задач указанного раздела, в которых прием срабатывал. Оказывается, что в нем всего пять задач. Нажимая клавишу "й", выясняем, что во второй задаче срабатывание приема было избыточным. Входим в трассировку и анализируем точку срабатывания. Прием пытался выразить u через z из уравнения $uz = 9$, в то время как уже имелась дизъюнкция $u = 9z \ \vee u = z/9$. Очевидно, такая попытка не нужна. Можно было бы отсечь ее, добавив фильтр, запрещающий срабатывание приема при наличии дизъюнкции уравнений.

   Чтобы выяснить, к чему приведет удаление фильтра "конец(альтернатива(...))", перекомпилируем прием с отброшенным фильтром и запускаем прокрутку подраздела "Решение уравнений" (в других подразделах срабатывание приема маловероятно, хотя для полной гарантии можно было бы реализовать и их тестирование). Итоги прокрутки таковы: 8 задач изменили свои ответы, одна замедлилась на 10 млн. шагов работы интерпретатора и еще несколько замедлились на меньшие величины. Общее замедление составило 20 млн. шагов. Нажимая "и" из диалогового блока, просматриваем изменившиеся ответы. В нескольких случаях они чуть-чуть усложнились, в нескольких - были переставлены их элементы. Видимо, наиболее чувствительной точкой оказалась сильно замедлившаяся задача. Нажимаем "з" и находим ее условие: $-y\sqrt{y}+x\sqrt{x}=a(-\sqrt{y}+\sqrt{x}), x^2+xy+y^2=b^2$. Параметры a, b положительны. Входим в трассировку и обнаруживаем, что прием применяется для выражения u через z из уравнения $uz=(a+b)(a-b)/2a$. При этом имеется второе уравнение $(u+z)^2=(3a^2-b^2)/2a$. Очевидно, в такой ситуации лучше было бы выразить u через z из второго уравнения, предварительно возведенного в степень 1/2.

10.10 Приемы символа "меньше"

Продолжаем рассмотрение приемов решателя, относящихся к элементарной алгебре. Переходим к неравенствам. Здесь приемы разбиты на две группы - для строгих и для нестрогих неравенств. Первые сгруппированы в разделе символа "меньше", вторые - в разделе символа "меньшеилиравно". Имеется небольшое пересечение, вызванное использованием указателей, позволяющих относить прием одновременно и к строгим и к нестрогим неравенствам (например, указателей "альтернатива"). Заметим, что символы "больше", "большеилиравно" допустимы для использования в задачах, однако они сразу же сводятся к символам "меньше", "меньшеилиравно". Это выполняют два простых приема, реализованных на ЛОСе. На ГЕНОЛОГе приемов для символов "больше", "большеилиравно" практически нет (исключение составляют несколько приемов, используемых при анализе текстов). Начнем со строгих неравенств.

Утверждение "меньше(a b)", означающее, что число a меньше числа b, прорисовывается формульным редактором в виде $a<b$. Справочники "предикатныйсимвол", "арность", "транзитивно", "одз" определяют простейшие свойства отношения "меньше". Справочник "монотонно" указывает, что истинность строгого неравенства сохраняется при монотонном неубывании второго операнда и монотонном невозрастании первого. Справочник "ослабление" определяет для символа "меньше" ослабленную версию предиката - символ "меньшеилиравно".

Стандартизация строгих неравенств

Для приведения строгих неравенств к стандартному виду используются следующие приемы:

1. Группировка всех ненулевых слагаемых в одной части неравенства.

   $\forall_{ab}(b<a \leftrightarrow 0<a-b)$

   Прием имеет заголовок "второйтерм". Выражения a, b отличны от константы 0. Срабатывание блокируется в следующих случаях:

   (a) Неравенство расположено в задаче на исследование либо в условии задачи на описание и содержит неизвестные. Исключение составляет случай, когда неравенство находится под отрицанием.

   (b) Редактируется ответ задачи на описание, причем уже выполнялось разрешение сопровождающих неравенств относительно параметров (последнее распознается по наличию комментария "стандменьше").

(c) Неравенство расположено в посылках задачи на доказательство, причем в одной его части расположена переменная, а в другой - константное выражение.

(d) Неравенство не расположено под квантором или описателем, связывающим какую-либо его переменную.

(e) Неравенство расположено в задаче на исследование, имеющей цель "анализфразы" либо "текстоваязадача". В последнем случае блокировка относится только к термам "меньше(...)", число операндов которых не равно двум.

Уровень срабатывания приема равен 0. Создан еще один прием группировки, применяемый в задачах на исследование: $\forall_{abcd}(a/b < c/b+d \leftrightarrow 0 < (c-a)/b+d)$. Он срабатывает на уровне 2.

2. Устранение минуса в одной части неравенства, если другая часть нулевая.

   $\forall_a(0 < -a \leftrightarrow a < 0)$

   $\forall_a(-a < 0 \leftrightarrow 0 < a)$

   Оба приема срабатывают на уровне 2. Первый из них может нарушить стандартизацию, на которую рассчитаны некоторые приемы для комплекснозначных отображений, тогда он блокируется.

3. Отрицание строгого неравенства. Такие отрицания сразу преобразуются в нестрогие неравенства:

   $\forall_{ab}(\neg(a < b) \leftrightarrow b \leq a)$

   Прием срабатывает на уровне 0.

4. Положительность либо отрицательность дроби. Утверждение преобразуется в эквивалентное утверждение для произведения:

$$\forall_{ab}(0 < \frac{a}{b} \leftrightarrow 0 < ab)$$

$$\forall_{ab}(\frac{a}{b} < 0 \leftrightarrow ab < 0)$$

   Преобразование блокируется в задачах на анализ текста. Указатель "удалениеусловия(условие(и(тип(описать)условие))не(равно(x2 0)))" обеспечивает удаление сопровождающего утверждения $\neg(b = 0)$ в условиях задачи на описание, если оно после преобразования становится ненужным для сопровождения по о.д.з. Указатель "сопровождение" в первом приеме разрешает его применение даже тогда, когда преобразуемое утверждение используется для сопровождения по о.д.з. Уровень срабатывания равен 0.

5. Перенесение нуля в левую часть, если имеются слагаемые со знаком "минус". Таким образом достигается дополнительная стандартизация неравенства. Начинают преобладать утверждения о положительности значений выражений.

   $\forall_{ab}(a - b < 0 \leftrightarrow 0 < b - a)$

   Уровень срабатывания равен 1.

6. Замена строгого неравенства на отрицание равенства, если из контекста усматривается нестрогое неравенство.

   $\forall_a(a \leq 0 \rightarrow \neg(a = 0) \leftrightarrow a < 0)$

   Прием применяется справа налево. Если решается задача на описание, причем текущее условие или вся задача имеют комментарий "стандменьше", то прием блокируется. Такой комментарий указывает, что уже выполнялось редактирование сопровождающих неравенств в ответе задачи. Антецедент обрабатывается проверочным оператором. Указатель "удалениепосылок(1 х2 и(тип(описать) цель(редакция) известно(корень)условие не(известно(х2))))" означает, что если редактируется ответ задачи на описание и текущее неравенство не содержит неизвестных, то при проверке антецедента будут отбрасываться все содержащие неизвестные утверждения контекста. Уровень срабатывания равен 3.

   $\forall_a(0 \leq a \rightarrow \neg(a = 0) \leftrightarrow 0 < a)$

   Этот прием аналогичен предыдущему. Он имеет дополнительные фильтры, блокирующие срабатывание в следующих случаях:

   (a) Задача имеет тип "преобразовать" либо "исследовать".

   (b) Задача типа "описать" имеет цель с заголовком "известно" либо "зависит".

   (c) Неравенство является посылкой задачи на доказательство.

   (d) Неравенство входит в условие задачи на описание, содержит неизвестные, причем выражение a имеет вид $\operatorname{tg} b$.

   (e) Неравенство входит в условие задачи на описание, имеющей цель "вариант", причем это условие выделено комментарием "разборслучаев".

7. Замена дизъюнкции противоположных строгих неравенств на отрицание равенства.

   $\forall_{ab}(a < b \ \vee \ b < a \leftrightarrow \neg(a = b))$

   Прием блокируется, если текущий терм задачи выделен комментарием "разборслучаев". Исключение составляют ситуации, когда этот терм является условием задачи на доказательство либо на преобразование. Указатель "дизъюнктоперанд" блокирует попытки применить прием к конъюнкции отрицаний неравенств. Аналогичный прием создан для двух неравенств, левые части которых равны 0, а правые суть разности, отличающиеся знаком:

   $\forall_{ab}(0 < a - b \ \vee \ 0 < b - a \leftrightarrow \neg(a - b = 0))$

   Уровни срабатывания обоих приемов равны 0.

8. Дизъюнкция строгого неравенства и равенства.

   $\forall_{ab}(a = b \ \vee \ a < b \leftrightarrow a \leq b)$

   Прием срабатывает без ограничений. Уровень срабатывания равен 2. Следующие два приема учитывают возможность одновременного изменения знака у всех слагаемых ненулевой части неравенства:

   $\forall_{ab}(a - b \leq 0 \leftrightarrow a - b = 0 \ \vee \ 0 < b - a)$

   $\forall_{ab}(a - b = 0 \ \vee \ b - a < 0 \leftrightarrow 0 \leq a - b)$

В первом случае замена выполняется справа налево, во втором - слева направо. Указатели "нормзнака(x2 минус плюс)" и "отрицание(минус x1)" определяют следующий порядок идентификации. Сначала рассматривается произвольное слагаемое ненулевой части неравенства. Если оно имеет заголовок "минус", то a идентифицируется с его первым операндом, иначе - со вспомогательным термом, полученным из слагаемого добавлением внешнего знака "минус". Затем b идентифицируется с суммой остальных слагаемых, у которых знаки изменены на обратные. Уровень срабатывания равен 2.

9. Дизъюнкция строгого и нестрогого неравенств.

   $\forall_{ab}(a < b \ \vee \ a \leq b \leftrightarrow a \leq b)$

   Прием срабатывает без ограничений. Уровень срабатывания равен 3.

10. Устранение множителя одной части неравенства, имеющего определенный знак, если другая часть неравенства нулевая.

    $\forall_{ab}(0 \leq a \rightarrow 0 < ab \leftrightarrow 0 < a \ \& \ 0 < b)$

    Указатель "дробь(фикс(0 1)фикс(0 2 2))" делает возможным применение приема к неравенству вида $ab < 0$. При этом заменяющее утверждение приобретает вид $0 < a \ \& \ b < 0$. Антецедент обрабатывается проверочным оператором, причем указатель "лимит(50000 x1)" ограничивает трудоемкость попытки усмотрения неотрицательности a. Ограничитель трудоемкости вставляется в программу приема после точки идентификации множителя a, так что на анализ каждого сомножителя отводится один и тот же лимит. Указатель "сопровождение" означает, что прием применяется даже к неравенствам, используемым для сопровождения по о.д.з. Наконец, указатель "внимание(x2)" приводит к понижению до 0 весов всех условий и посылок, содержащих выражение b. Это может привести к срабатыванию каких-либо других приемов, проверяющих положительность данного выражения. Прием имеет два уровня срабатывания - 0 и 2. Имеется аналогичный прием для проверки неположительности множителя a:

    $\forall_{ab}(a \leq 0 \rightarrow 0 < ab \leftrightarrow a < 0 \ \& \ b < 0)$

    Единственное отличие - отсутствие указателя "внимание(x2)". Рассмотренные приемы анализировали знак отдельных сомножителей ненулевой части неравенства. Если в контексте неравенства имеется другое неравенство, явно указывающее знак некоторого подпроизведения ненулевой части, то применяются другие два приема. Их теоремы такие же, как и выше:

    $\forall_{ab}(0 \leq a \rightarrow 0 < ab \leftrightarrow 0 < a \ \& \ 0 < b)$

    $\forall_{ab}(a \leq 0 \rightarrow 0 < ab \leftrightarrow a < 0 \ \& \ b < 0)$

    Антецеденты здесь не обрабатываются проверочным оператором, а берутся непосредственно из контекста. Фильтр "заголовок(x1 умножение)" ограничивает рассмотрение только таких неравенств для a, у которых ненулевая часть имеет вид произведения. Указатель "вариант(фикс(1)меньше)" разрешает брать из контекста не только нестрогие, но и аналогичные строгие неравенства. Прочие указатели приемов - "дробь(фикс(0 1)фикс(0 2 2))", "внимание(x2)" (только у первого приема), "сопровождение" - уже были прокомментированы выше. Если ненулевая часть неравенства имеет вид суммы, то предпринимается попытка разложить ее на множители и после этого найти сомножитель определенного знака:

$\forall_{abc}(a = bc \ \& \ 0 \leq b \to 0 < a \leftrightarrow 0 < c)$

$\forall_{abc}(a = bc \ \& \ 0 \leq b \to a < 0 \leftrightarrow c < 0)$

Здесь a - выражение с заголовком "плюс". Первый антецедент выделен указателем "идентификатор". Он предпринимает попытку разложения a на множители с помощью нормализатора "факторизация". При этом используются лишь простейшие приемы, в основном - вынесение за скобку общего множителя всех слагаемых. Второй антецедент обрабатывается проверочным оператором. В задачах на исследование, имеющих цель "известно", приемы блокируются. Уровень срабатывания равен 4.

11. Декомпозиция неравенства с произведением в одной части и нулем в другой, если известен знак суммы либо разности сомножителей.

    $\forall_{abc}(0 \leq a + c \to 0 < (a + b)(c - b) \leftrightarrow 0 < a + b \ \& \ 0 < c - b)$

    $\forall_{abc}(0 \leq b - c \to (a + b)(a + c) < 0 \leftrightarrow 0 < a + b \ \& \ a + c < 0)$

    Первый прием имеет указатель "нормзнака(x2 минус плюс)", позволяющий идентифицировать $-b$ как сумму всех слагаемых выражения b, взятых с обратными знаками. Уровень срабатывания приемов равен 0.

12. Усмотрение двух противоположных неравенств.

    $\forall_{ab}(a < b \to \neg(b < a))$

    Прием имеет заголовок "второйтерм". Антецедент идентифицируется с неравенством, имеющимся в контексте текущего противоположного неравенства. Тогда последнее заменяется на константу "ложь". Указатель "альтернатива(фикс(1) меньшеилиравно)" разрешает идентифицировать антецедент с нестрогим неравенством. При текстовом анализе прием блокируется. Уровень срабатывания равен 0.

13. Замена неравенства с модулем на два неравенства.

    $\forall_{abc}(0 \leq c \to 0 < a + c|b| \leftrightarrow 0 < a + bc \ \vee \ 0 < a - bc)$

    Неравенство должно являться посылкой задачи на доказательство. Уровень срабатывания равен 4.

    $\forall_{abc}(c \leq 0 \to 0 < a + c|b| \leftrightarrow 0 < a + bc \ \& \ 0 < a - bc)$

    Неравенство является посылкой либо условием задачи, не используемой для сопровождения по о.д.з. Если тип задачи - "доказать", причем b содержит конечную сумму, конечное произведение либо модуль, то уровень срабатывания достаточно большой (равен 7). Обычно в такой ситуации применяются другие приемы, позволяющие получить оценку для b. Если данная ситуация не имеет места, то в посылках прием срабатывает на уровне 1, а в условии задачи на доказательство - на уровне 4. К условиям задач на описание прием не применяется. В задачах на исследование, имеющих цель "известно", прием применяется только при отсутствии в неравенстве невырожденных числовых атомов.

14. Исключение модуля в неравенстве.

    $\forall_{ab}(a + b < 0 \to a + |b| < 0 \leftrightarrow a - b < 0)$

Прием применяется без ограничений. Уровень срабатывания равен 1.

15. Попытка разложения на множители ненулевой части неравенства для параметров.

    $\forall_{ab}(b = a \to 0 < a \leftrightarrow 0 < b)$

    Антецедент обращается к нормализатору "видумножение" с ослабляющим попытки разложения комментарием "нормуравнение". Неравенство представляет собой условие задачи на описание, не используется для сопровождения по о.д.з. и не содержит неизвестных. Фильтр "не(блокнеравенства)" проверяет отсутствие в задаче уравнений, имеющих общие переменные с рассматриваемым неравенством. Указатель "дробь" разрешает применять прием к неравенствам с нулем в правой части. Уровень срабатывания равен 3.

16. Символы бесконечности. Несколько приемов усматривают истинность неравенств с символами бесконечности. Такие неравенства могут появляться в специальных контекстах.

    $-\infty < \infty$

    $\forall_a(a - \text{число} \to -\infty < a)$

    $\forall_a(a - \text{число} \to a < \infty)$

    $\forall_a(\neg(\infty < a))$

    $\forall_a(\neg(a < -\infty))$

    Приемы имеют заголовок "второйтерм". Их уровень срабатывания равен 0.

17. Переход от строгого неравенства для целочисленного выражения к нестрогому. Такие переходы являются общей тенденцией в решателе.

    $\forall_n(n - \text{целое} \to 0 < n \leftrightarrow 0 \leq n - 1)$

    Неравенство является посылкой задачи. Должна также существовать посылка с заголовком "целое" либо "натуральное" либо "четное" - таким образом отсекаются попытки применения приема в тех задачах, где нет явных указаний на рассмотрение целочисленных значений. Преобразуемое неравенство не должно являться индуктивным предположением при доказательстве по индукции. В задачах на исследование, имеющих цель "известно", прием применяется на уровне 3, иначе - на уровне 1. Для той же теоремы создан еще один прием. Он применяется к неравенству, расположенному под конечными суммой либо произведением и срабатывает на уровне 0. Таким образом достигается стандартизация условий на область суммирования (перемножения).

    $\forall_n(n - \text{целое} \to n < 0 \leftrightarrow n = 1 \leq 0)$

    Прием аналогичен предыдущему. Уровень срабатывания равен 1.

18. Переход от строгого неравенства для целочисленного переменного к равенству.

    $\forall_{abn}(n - \text{целое}\ \&\ 0 \leq n + a\ \&\ a + b - 1 = 0 \to 0 < b - n \leftrightarrow n = -a)$

    Здесь второй антецедент берется непосредственно из контекста неравенства; a, b - целочисленные константы. Первый антецедент обрабатывается проверочным оператором, последний - выделен указателем "идентификатор". Уровень срабатывания равен 0.

$\forall_{abx}(a \leq x \ \& \ x - \text{целое} \ \& \ b - a = 1 \rightarrow x < b \leftrightarrow x = a)$

Аналогично предыдущему. Из контекста берется первый антецедент.

$\forall_{amnk}(a \in \{m, \ldots, n\} \ \& \ k - m - 1 = 0 \rightarrow a < k \leftrightarrow a = m)$

Первый антецедент берется из конекста неравенства, второй выделен указателем "идентификатор". Уровень срабатывания равен 1.

19. Усмотрение двух возможных значений целочисленного выражения.

    $\forall_n(n - \text{целое} \ \& \ 0 \leq n \rightarrow 0 < 2 - n \leftrightarrow n = 0 \ \vee \ n = 1)$

    Неравенство представляет собой условие условного выражения. Уровень срабатывания равен 0.

20. Использование тождества для перехода к неравенству с одной переменной.

    $\forall_{abcdef}(d - a - b = 0 \ \& \ c + d = e \ \& \ f = (0 < e) \rightarrow 0 < a + b + c \leftrightarrow f)$

    Первый антецедент идентифицируется с равенством из контекста. Неравенство входит в посылку задачи на исследование и имеет не менее двух свободных переменных. Второй антецедент, выделенный указателем "идентификатор", обрабатывает сумму $c + d$ ненулевых частей равенства и неравенства нормализатором "видумножение". Третий антецедент, тоже выделенный указателем "идентификатор", обрабатывает неравенство для данной суммы нормализатором стандартизации строгих неравенств "нормменьше". Проверяется, что результат f имеет единственную свободную переменную. Уровень срабатывания равен 2.

21. Исключение логарифма.

    $\forall_{axy}(0 < a - 1 \rightarrow \log_a x < \log_a y \leftrightarrow x < y)$

    $\forall_{axy}(a - 1 < 0 \rightarrow \log_a x < \log_a y \leftrightarrow y < x)$

    Приемы срабатывают без ограничений. Уровень срабатывания равен 1.

22. Сокращение обеих частей неравенства на положительный множитель. Приемы применяются в задачах на исследование, имеющих цель "известно", так как здесь заблокировано перенесение всех ненулевых членов в одну часть.

    $\forall_{abc}(0 < a \rightarrow ab < ac \leftrightarrow b < c)$

    $\forall_{abc}(0 < a \rightarrow 0 < ac + ab \leftrightarrow 0 < b + c)$

    Первый прием срабатывает на уровне 3, второй - на уровне 2.

23. Усмотрение истинности дизъюнкции двух строгих неравенств.

    $\forall_{ab}(0 < ab \rightarrow a < 0 \ \vee \ 0 < b)$

    Прием имеет заголовок "второйтерм". Он применяется в условии задачи на описание. Уровень срабатывания равен 2.

Усмотрение истинности неравенства при помощи оператора "усмменьше"

Для усмотрения истинности строгих неравенств используется проверочный оператор "усмменьше". Описание его приемов будет дано ниже, здесь же ограничимся приемами, использующими данный оператор. Их теорема имеет вид:

$\forall_{ab}(a < b \rightarrow a < b)$

Антецедент обрабатывается проверочным оператором, все приемы имеют заголовок "второйтерм". Они отличаются друг от друга лишь условиями на контекст срабатывания. Начнем с версии приема, применяемой в наиболее общем случае. Фильтры этой версии таковы:

1. Если решается задача на доказательство, то отсутствует комментарий (нижняягрань x). Для этой ситуации создана другая версия приема, имеющая меньший уровень срабатывания (см.ниже). Таким образом обеспечивается попытка быстрой проверки перед применением трудоемкого альтернативного способа доказательства (дифференцирования по переменной x).

2. Если неравенство входит в условие задачи на описание, зависящее от неизвестных, то уровень срабатывания равен 4, иначе он равен 2.

3. Неравенство не используется для сопровождения по о.д.з.

4. Если решается задача на описание, то отсутствует комментарий "стандменьше". Такой комментарий означает, что имеет место этап явного разрешения неравенств для параметров при редактировании ответа. Аналогично, если неравенство входит в условие задачи на описание, то это условие не имеет комментария "стандменьше".

5. Если редактируется ответ задачи на описание, причем рассматриваемое неравенство является условием задачи либо расположено в ее условии под квантором существования, то неверно, что одна из частей неравенства представляет собой неизвестную, а другая часть известна.

6. Не решается задача на преобразование.

7. Неравенство не расположено в посылке, полученной при выводе следствий.

8. Если решается задача на исследование, то неравенство не имеет вида $0 < b$, где b - переменная. Такие неравенства полезны, даже если они вытекает из других посылок.

9. Не решается задача на исследование, имеющая цель "известно".

10. Если неравенство входит в посылку задачи на описание, то эта задача не имеет цели (известно ...) либо (независит ...).

Введен сильный ограничитель трудоемкости. Прочие версии приема предназначены для следующих ситуаций:

1. Редактируется ответ задачи на описание, причем неравенство является условием задачи и не содержит неизвестных. Задача и рассматриваемое условие не имеют комментария "стандменьше". Прием применяется, даже если неравенство используется для сопровождения по о.д.з. Уровень срабатывания равен 4.

2. Неравенство входит в условие задачи на доказательство, имеющей комментарий (нижняягрань x). Уровень срабатывания равен 1.

3. Неравенство входит в дизъюнктивное условие задачи на описание и не имеет переменных, связанных внешними кванторами либо описателями. Успешное применение приема иногда позволяет исключить дизъюнкцию. Уровень срабатывания равен 1.

Усмотрение ложности неравенства при помощи оператора "усменьшеилиравно"

Для усмотрения истинности нестрогих неравенств используется проверочный оператор "усменьшеилиравно". Его приемы будут описаны ниже, здесь же приведем прием, обращающийся к оператору для усмотрения ложности строгого неравенства.

$\forall_{ab}(b \leq a \rightarrow \neg(a < b))$

Прием имеет заголовок "второйтерм"; он заменяет строгое неравенство $a < b$ на константу "ложь". Антецедент обрабатывается проверочным оператором. Введен сильный ограничитель трудоемкости, делающий попытку применения приема быстрой. Если решается задача на описание и текущее условие содержит неизвестные, то уровень срабатывания равен 4, иначе он равен 3. Имеются следующие дополнительные ограничения на срабатывание (они встречались у приемов, предпринимающих попытку усмотрения истинности неравенства):

1. Если решается задача на описание, то отсутствует комментарий "стандменьше".

2. Не решается задача на преобразование.

3. Если редактируется ответ задачи на описание, причем рассматриваемое неравенство является условием задачи либо расположено в ее условии под квантором существования, то неверно, что одна из частей неравенства представляет собой неизвестную, а другая часть известна.

4. Не решается задача на исследование, имеющая цель "известно".

Еще одна версия приема применяется к неравенствам, входящим в дизъюнктивные условия задач на описание. Она имеет уровень срабатывания 1.

Непосредственное усмотрение ложности строгих неравенств

$\forall_{ab}(0 < a - b \rightarrow \neg(0 < b - a))$

Прием имеет заголовок "второйтерм". Он усматривает в контексте вхождение неравенства $0 < a - b$ и заменяет неравенство $0 < b - a$ на константу "ложь". Попытка применения приема предпринимается при усмотрении неравенства, входящего в условие задачи на описание внутри дизъюнкции. Уровень срабатывания равен 1.

Использование оператора "провменьше" для доказательства строгих неравенств

Кроме проверочного оператора "усменьше", для проверки строгих неравенств создан еще один оператор - "провменьше". Он существенно усиливает возможности первого оператора, но работает медленнее. Поэтому по умолчанию компилятор использует в приемах первый оператор. Приемы оператора "провменьше" будут приведены ниже, здесь же ограничимся приемом, обращающимся к данному оператору:

$\forall_{ab}(a < b \to a < b)$

Прием имеет заголовок "второйтерм". Его антецедент выделен указателем "проверка", означающим применение усиленной версии проверочного оператора. Неравенство является условием задачи на доказательство, причем максимальный уровень этой задачи должен быть больше 4. Перед обращением к проверочному оператору выполняется эвристическая оценка возможности доказательства неравенства элементарными средствами, реализуемая оператором фильтра "фильтрнеравенства". Последний содержится в подразделе базы приемов, посвященном доказательству неравенств с помощью производной. Он блокирует попытку применения приема при выполнении любого из следующих требований:

1. Некоторая переменная имеет в неравенстве единственное алгебраическое вхождение (достижимое из корня через операции "плюс", "минус", "умножение", "дробь", "модуль", основания степеней и отношения "равно", "меньше", "меньшеилиравно"), а также вхождение, не являющееся алгебраическим.

2. Некоторая переменная имеет в неравенстве единственное вхождение под тригонометрической операцией, а также имеет вхождение не под тригонометрической операцией.

3. Некоторая переменная имеет в неравенстве единственное вхождение под одной из операций "логарифм", "арксинус", "арктангенс", "арккосинус", а также имеет вхождение не под этой же операцией.

4. Некоторая переменная имеет в неравенстве единственное вхождение в показателе стпени, а также имеет вхождение не в показателе степени.

Указатель "ключ(провменьше)" блокирует повторные попытки применения приема. Уровень срабатывания равен 3.

Неравенства под кванторами и описателями

Для неравенств, переменные которых связаны внешними кванторами или описателями, предусмотрены приводимые ниже приемы. Обычно подкванторное выражение удается сильно упростить другими приемами, так что рассматриваемые здесь приемы ориентированы на простейшие стандартные случаи, возникающие после преобразований.

1. Исключение квантора существования. Приемы усматривают истинность квантора, заменяя его на константу "истина", либо заменяют квантор на бескванторное утверждение, либо устраняют часть связанных переменных квантора. Сразу отметим, что приведенный список приемов никоим образом не претендует на полноту. В нем собраны всего лишь те случаи, которые понадобились для рассматривавшихся при обучении решателя задач.

   $\forall_{abd}(\exists_c(c < b \;\&\; a < c \;\&\; c - \text{число} \;\&\; d) \leftrightarrow a < b \;\&\; d)$

   Переменная c не входит в термы a, b, d. Указатель "альтернатива(фикс(0 1 2 2)меньшеилиравно)" разрешает вместо строгого неравенства $a < c$ идентифицировать нестрогое. Указатель "единица(истина x4)" разрешает отсутствие дополнительных утверждений d. Следующие два приема выполняют аналогичную бескванторную переформулировку:

$\forall_{abde}(\exists_c(c < b \; \& \; a < c \; \& \; c -$ число $\& \; \neg(c = e) \; \& \; d) \leftrightarrow a < b \; \& \; d)$

$\forall_a(\exists_n(n -$ натуральное $\& \; n < a) \leftrightarrow 1 < a)$

Уровень срабатывания всех этих приемов равен 0. Для усмотрения существования числа заданного типа, большего либо меньшего другого числа, служат следующие приемы, тоже срабатывающие на уровне 0:

$\forall_a(a -$ число $\rightarrow \exists_b(b -$ число $\& \; a < b))$

$\forall_a(a -$ число $\rightarrow \exists_b(b -$ число $\& \; b < a))$

$\forall_{ac}(a -$ число $\rightarrow \exists_b(b -$ число $\& \; a < b \; \& \; \neg(b = c)))$

$\forall_{ac}(a -$ число $\rightarrow \exists_b(b -$ число $\& \; b < a \; \& \; \neg(b = c)))$

$\forall_a(a -$ число $\rightarrow \exists_b(b < a))$

$\forall_a(a -$ число $\rightarrow \exists_b(a < b))$

$\forall_a(a -$ число $\rightarrow \exists_n(n -$ натуральное $\& \; a < n))$

$\forall_a(a -$ число $\rightarrow \exists_n(n -$ целое $\& \; a < n))$

$\forall_a(a -$ число $\rightarrow \exists_n(n - \mathrm{rational} \; \& \; a < n))$

Наконец, приведем приемы, исключающие одну из переменных кванторной приставки.

$\forall_{fg}(\exists_{xy}(x -$ число $\& \; x < f(y) \; \& \; g(y)) \leftrightarrow \exists_y(g(y)))$

Указатель "отображение(x6 x7)" определяет функциональные переменные f, g, т.е. разрешает идентифицировать $f(y), g(y)$ с произвольными термами. Указатель "дробь(фикс(0 1 3 2))" разрешает переставлять при идентификации части неравенства. Указатель "кортежпеременных(x24)" означает, что y идентифицируется со списком всех отличных от x переменных кванторной приставки. Вообще говоря, этот список может быть пустым, однако фильтр "не(заголовок(x24 пустоеслово))" требует, чтобы в нем была хотя бы одна переменная. Уровень срабатывания равен 0.

$\forall_{axP}(a(z) - \mathrm{rational} \; \& \; \neg(a(z) = 0) \rightarrow \exists_{yz}(0 < y \; \& \; y - \mathrm{rational} \; \& \; x = a(z)y \; \& \; P(z)) \leftrightarrow \exists_z(P(z) \; \& \; x - \mathrm{rational} \; \& \; 0 < a(z)x))$

Переменные a, P - функциональные. Антецеденты обрабатываются проверочными операторами, к списку посылок которых добавляются конъюнктивные члены утверждения $P(z)$. Уровень срабатывания равен 2.

2. Исключение квантора общности. Здесь представлены только приемы, выполняющие бескванторную переформулировку.

$\forall_{ab}(\forall_c(c \leq a \; \& \; c -$ число $\rightarrow c < b) \leftrightarrow a < b)$

$\forall_{ab}(\forall_c(a \leq c \; \& \; c -$ число $\rightarrow b < c) \leftrightarrow b < a)$

$\forall_{ab}(\forall_c(a \leq c \; \& \; c -$ число $\rightarrow b \leq c) \leftrightarrow b \leq a)$

$\forall_{ab}(\forall_c(c \leq a \; \& \; c -$ число $\rightarrow c \leq b) \leftrightarrow a \leq b)$

В последних двух приемах указатели "альтернатива(фикс(0 1 3)меньше)", "альтернатива(фикс(0 1 6)меньше)" разрешают идентификацию каждого из неравенств левой части со строгим неравенством, причем фильтр "или(заголовок(фикс(0 1 3)меньше)заголовок(фикс(0 1 6)меньшеилиравно))" запрещает идентификацию неравенства в антецеденте с нестрогим, а консеквента - со строгим неравенством. Уровни срабатывания всех перечисленных приемов равны 0.

$\forall_{abc}(\forall_x(x -$ число $\& \neg(x - a = 0) \; \& \; |a - x| < b \rightarrow |a - x| < c) \leftrightarrow b \leq c)$

Прием срабатывает на уровне 1.

$\forall_{abf}($возрастает$(\lambda_x(f(x), x -$ число$), [a, \infty)) \; \&$ одз$(f(a)) \rightarrow \forall_x(a < x \; \& \; x -$ число $\rightarrow b < f(x)) \leftrightarrow b \leq f(a))$

$\forall_{abf}($убывает$(\lambda_x(f(x), x -$ число$), [a, \infty)) \; \&$ одз$(f(a)) \rightarrow \forall_x(a < x \; \& \; x -$ число $\rightarrow f(x) < b) \leftrightarrow f(a) \leq b)$

В этих приемах переменная f выделена указателем "отображение", т.е. $f(x)$ идентифицируется с произвольным выражением. Первый антецедент, выделенный указателем "блокпроверок", проверяет, что выражение $f(x)$ монотонно убывает либо монотонно возрастает при $x \leq a$. Для этого используются проверочные операторы, которые будут описаны в разделе, посвященном математическому анализу. Второй антецедент, выделенный указателем "легковидеть", проверяет, что выражение $f(a)$ имеет смысл. Для этого решается вспомогательная задача на доказательство утверждений, описывающих о.д.з. данного выражения. Уровень срабатывания равен 2. В целочисленном случае создан следующий прием:

$\forall_{mn}(0 \leq n - 1 \rightarrow \forall_x(x -$ натуральное $\& \; x \leq m \rightarrow x < n) \leftrightarrow [m] < n)$

Его уровень срабатывания тоже равен 2. Если антецедент представляет собой условие принадлежности, а консеквент - неравенство, кванторная импликация дает условие включения в интервал:

$\forall_{Ab}(\forall_x(x \in A \rightarrow x -$ число $\& \; x < b) \leftrightarrow A \subseteq (-\infty, b))$

Уровень срабатывания равен 3.

3. Перенесение неравенства в консеквент.

   $\forall_{fgAB}(\forall_x(A(x) \; \& \; f(x) < g(x) \rightarrow \neg B(x)) \leftrightarrow \forall_x(A(x) \; \& \; B(x) \rightarrow g(x) \leq f(x)))$

   Переменные A, B, f, g выделены указателем "отображение", переменная x - указателем "кортежпеременных". Срабатывание блокируется в следующих случаях:

   (a) $B(x)$ имеет вид $y = t$, где y - связанная переменная внешнего описателя "класс".

   (b) Одно из выражений $f(x), g(x)$ представляет собой переменную, не входящую в другое выражение.

   Если решается задача на доказательтство, и утверждение $B(x)$ является равенством, то прием срабатывает на уровне 2, иначе - на уровне 4.

4. Использование кванторного строгого неравенства для усмотрения отрицания равенства.

   $\forall_{afg}(f(a) \; \& \; \forall_x(f(x) \rightarrow 0 < g(x)) \rightarrow \neg(g(a) = 0))$

   Прием имеет заголовок "второйтерм". Переменная g не выделена указателем "отображение", т.е. обозначает некоторую функцию. Фильтр "отр" означает, что утверждение $\neg(g(a) = 0)$ является посылкой либо условием задачи. Второй антецедент идентифицируется с кванторной импликацией из текущего контекста. Переменная f выделена указателем "отображение", т.е. $f(x)$ идентифицируется с конъюнкцией антецедентов. Первый антецедент выделен указателем "следствие" - он проверяется с помощью вспомогательной задачи на доказательство. Уровень срабатывания равен 3.

5. Сведение кванторной импликации с неравенством в консеквенте к условию включения образа в интервал.

   $\forall_{abfg}(\forall_x(x-\text{число}\ \&\ f(x) \to |g(x)+a| < b) \leftrightarrow \text{образ}(\lambda_x(g(x), x-\text{число}), \text{set}_x(x-\text{число}\ \&\ f(x))) \subseteq (-a-b, -a+b))$

   Так как вместо квантора общности возникают два описателя - "отображение" и "класс", прием не относится к категории упрощающих. Он применяется в специальных случаях - кванторная импликация должна являться условием задачи на описание, имеющей цель "пример", выражения $a, b, g(x)$ не должны содержать неизвестных, а выражение $f(x)$ должно их содержать. Переменные f, g выделены указателем "отображение". Уровень срабатывания равен 8. Фактически прием решателем не используется.

6. Раскрывание скобок. Иногда раскрывание скобок в выражениях, определяющих границы промежутка, позволяет усмотреть стандартный вид этого промежутка, используемый при исключении квантора или описателя. Раскрывание скобок выполняется следующим приемом:

   $\forall_{abcdenx}(a(b+c)+d=e \to x < a(b+c)+d \leftrightarrow x < e)$

   Антецедент обращается к нормализатору "стандплюс". Неравенство должно быть расположено под квантором существования либо под описателем "класс", причем переменная x должна связываться эти квантором или описателем. Неравенство должно входить в состав конъюнкции, содержащей также встречное строгое неравенство для x. Уровень срабатывания равен 2.

7. Решение неравенства относительно переменной, по которой выполняется суммирование либо перемножение. Если область суммирования либо перемножения задана с использованием неравенств, не разрешенных явно относительно варьируемой переменной, то для линейных неравенств такое разрешение выполняется:

   $\forall_{abci}(0 < b \to a+bi < c \leftrightarrow i < (c-a)/b)$

   $\forall_{abci}(b < 0 \to a+bi < c \leftrightarrow (c-a)/b < i)$

   Неравенство расположено под описателем "отображение", в связывающую приставку которого входит переменная i. Этот описатель расположен под операцией "суммавсех" либо "произведениевсех". Переменная i не входит в выражения a, b, c. Уровень срабатывания равен 3.

Решение неравенств

В дополнение к приемам сканирования задачи, при решении неравенств используются также ускоряющие пакеты "уравнменьше", "уравнменьшеилиравно". В них собраны несколько десятков несложных приемов, дублирующих приемы сканирования. Инициирует обращение к пакету прием сканирования, делающий первый шаг цепочки преобразований отдельного неравенства, в то время как пакет выполняет остальные шаги. Наиболее эффективен этот принцип при работе с дизъюнктивно-конъюнктивными логическими конструкциями, возникающими при рассмотрении подслучаев. Почти весь анализ подслучаев оказывается вынесен из сканирования, что ускоряет вычисления во много раз. Ускоряющие пакеты будут описаны ниже, здесь же перечислим приемы сканирования:

1. Решение неравенств $-x < a$, $a < -x$.

   $\forall_{ab}(a < -b \leftrightarrow b < -a)$

   $\forall_{ab}(-b < a \leftrightarrow -a < b)$

   Приемы срабатывают в условии задачи на описание либо в задаче на исследование. Допустимые заголовки надутверждений неравенства - "и", "или", "не", "существует". Выражение b содержит неизвестные, a - не содержит. Если a - переменная, связанная внешним квантором либо описателем "класс", то прием блокируется. Уровень срабатывания равен 1.

   На тех же теоремах основаны еще два приема, в которых b - неконстантное выражение, a - константа. Наличие неизвестных этими приемами игнорируется. Уровни срабатывания равны 1. Один из приемов блокируется в специальных ситуациях, встречающихся в задачах на комплекснозначные функции. Заметим, что перечисленные приемы не использовали ускоряющих пакетов решения неравенств.

2. Решение неравенств $x + a < b, b < x + a$.

   $\forall_{abc}(b < a + c \leftrightarrow b - c < a)$

   $\forall_{abc}(a < b - c \leftrightarrow a + c < b)$

   Первый прием выполняет замену слева направо, второй - справа налево. Ограничения на контекст те же, что в предыдущем пункте: неравенство является условием задачи на описание либо посылкой задачи на исследование, допустимые внешние символы - "и", "или", "не", "существует". Выражение a содержит неизвестные, выражения b, c - не содержат. При этом a идентифицируется как совокупность всех неизвестных слагаемых соответствующей части неравенства. Неравенство не используется для сопровождения по о.д.з. Приемы блокируются, если в задаче решаются уравнения, а неравенства играют лишь второстепенную роль. Тогда учет неравенств будет иметь характер "проверки" после подстановки в них найденных значений неизвестных. Усмотрение второстепенной роли неравенств обеспечивается фильтром "блокнеравенства", проверяющим наличие уравнений - корневых либо внутри дизъюнкций. В специальных случаях (решение задачи на отыскание корней производной, наличие параметрического описания для неизвестной, входящей в неравенство) данная блокировка отменяется. Кроме того, приемы блокируются, если b - переменная, связанная внешним квантором или описателем "класс", а также при текстовом анализе. Уровень срабатывания равен 1.

3. Решение неравенств $xa < b, b < xa$.

   $\forall_{abc}(b < ac \leftrightarrow c < 0 \;\&\; a < b/c \;\vee\; 0 < c \;\&\; b/c < a \;\vee\; c = 0 \;\&\; b < 0)$

   $\forall_{abc}(ac < b \leftrightarrow c < 0 \;\&\; b/c < a \;\vee\; 0 < c \;\&\; a < b/c \;\vee\; c = 0 \;\&\; 0 < b)$

   Неравенство входит в условие задачи на описание либо в посылку задачи на исследование. Допустимые заголовки надтермов - "и", "или", "не", "существует". Выражение a содержит неизвестные, выражения b, c - не содержат. При этом указатель "перечень(x1 не(известно(x1)))" определяет идентификацию a

как произведение всех неизвестных сомножителей соответствующей части неравенства. Неравенства $a < b/c, b/c < a$ обрабатываются нормализатором "уравнменьше", который в простейших случаях доводит их решение до конца. Например, таким образом решаются неравенства с произведением линейных относительно неизвестной множителей в одной части и известной другой частью. В частности, это накрывает метод интервалов. При обращении к нормализатору "уравнменьше" ему передается в качестве дополнительной посылки сопровождающее неравенство для c. Последнее обрабатывается лишь нормализатором общей стандартизации "нормменьше". Условие $c = 0$ тоже обрабатывается нормализатором общей стандартизации "нормчисло". Таким образом, сразу отсекаются заведомо невыполнимые подслучаи. Весь заменяющий терм обрабатывается нормализатором "нормлог", учитывающим обращение подтермов в логические константы "истина", "ложь". Поэтому, если удастся усмотреть определенный строгий знак выражения c, заменяющий терм сохранит лишь единственный подслучай. Чтобы упростить работу перечисленных нормализаторов, перед обращением к ним выражения b, c обрабатываются нормализатором "видумножение".

4. Группировка в одной части всех слагаемых и последующее преобразование неравенства. Прием применяется, если обе части неравенства содержат неизвестные:

   $\forall_{ab}(a < b \leftrightarrow 0 < b - a)$

   Неравенство входит в условие задачи на описание либо в посылку задачи на исследование. Допустимые заголовки надтермов - "и", "или", "существует". Выражения a, b содержат неизвестные. При редактировании ответа прием блокируется, если одна из частей неравенства представляет собой неизвестную, не входящую в другую часть. Если неравенство содержит степени с неизвестными показателями, то удаляются комментарии, блокирующие повторную попытку перехода к новым неизвестным. Заменяющее неравенство обрабатывается нормализатором "уравнменьше", который во многих случаях обеспечивает его явное разрешение относительно неизвестных. Уровень срабатывания равен 1.

5. Положительность либо отрицательность произведения. Происходит разбиение неравенства на группу более простых неравенств:

   $\forall_{ab}(ab < 0 \leftrightarrow a < 0 \,\&\, 0 < b \;\vee\; 0 < a \,\&\, b < 0)$

   $\forall_{ab}(0 < ab \leftrightarrow a < 0 \,\&\, b < 0 \;\vee\; 0 < a \,\&\, 0 < b)$

   Преобразуемое неравенство является условием задачи на описание и не используется для сопровождения по о.д.з. Задача не имеет уравнений. Сомножитель a содержит неизвестные. Неравенства $a < 0, 0 < a$ обрабатываются нормализатором "уравнменьше", неравенства $0 < b, b < 0$ - либо этим же нормализатором, либо (при отсутствии в них неизвестных) нормализатором общей стандартизации "нормменьше". Вся заменяющая дизъюнкция обрабатывается нормализаторами "склейканеравенств" и "нормлог". Первый из них предназначен для упрощения дизъюнктивно-конъюнктивных логических конструкций с неравенствами и будет описан ниже. Приемы срабатывают на уровне 2. В случае задач на исследование созданы дополнительные версии указанных двух приемов. Они срабатывают только при наличии цели "линия", указывающей на исследование свойств линии, заданной своим уравнением. Заменяющая дизъюнкция снабжается комментарием "разборслучаев". Уровень срабатывания равен 5.

6. Устранение дробной части неравенства с неизвестными. Неравенства для дроби преобразуются в неравенства для произведения:

   $\forall_{abc}(b - \text{число}\ \&\ \neg(b = 0) \to c < a/b \leftrightarrow 0 < b\ \&\ bc < a\ \vee\ b < 0\ \&\ a < bc)$

   $\forall_{abc}(b - \text{число}\ \&\ \neg(b = 0) \to a/b < c \leftrightarrow 0 < b\ \&\ a < bc\ \vee\ b < 0\ \&\ bc < a)$

   Неравенство должно входить в условие задачи на описание либо в посылку задачи на исследование. Либо a, либо b содержит неизвестные, c - не содержит неизвестных. Допустимые заголовки надтермов - "и", "или". Задача не имеет уравнений. Неравенства $a < bc$, $bc < a$ обрабатываются нормализатором "уравнменьше", неравенства $0 < b, b < 0$ - нормализаторами "нормусм" и "уравнменьше" либо (при отсутствии неизвестных) "нормменьше". Вся заменяющая конструкция обрабатывается нормализаторами "склейканеравенств" и "нормлог". Уровни срабатывания равны 1 и 3, причем срабатывание второго приема на уровне 1 при наличии в неравенстве логарифмов с различными основаниями блокируется.

7. Объединение двух односторонних неравенств. По мере того, как отдельные неравенства задачи явно разрешаются относительно неизвестной, возникает необходимость объединения двух однонаправленных неравенств, ограничивающих диапазон изменения одной и той же неизвестной. Вообще говоря, такое объединение приводит к разбору случаев.

   $\forall_{abc}(a \leq c\ \&\ b < c \leftrightarrow 0 < a - b\ \&\ a \leq c\ \vee\ \neg(0 < a - b)\ \&\ b < c)$

   $\forall_{abc}(c \leq a\ \&\ c < b \leftrightarrow a - b < 0\ \&\ c \leq a\ \vee\ \neg(a - b < 0)\ \&\ c < b)$

   $\forall_{abc}(c < a\ \&\ c < b \leftrightarrow 0 \leq a - b\ \&\ c < b\ \vee\ \neg(0 \leq a - b)\ \&\ c < a)$

   $\forall_{abc}(a < c\ \&\ b < c \leftrightarrow 0 \leq a - b\ \&\ a \leq c\ \vee\ \neg(0 \leq a - b)\ \&\ b < c)$

   Приемы имеют заголовок "заменаусловия(второйтерм)", каждый из них заменяет два условия задачи на описание на одно новое утверждение. Либо выражение c содержит неизвестные, а выражения a, b не содержат, либо выражение c отлично от нуля, а выражения a, b константные. Неравенства не используются для сопровождения по о.д.з. Задача не имеет уравнений, переменные которых входят в преобразуемые неравенства. Если кроме двух преобразуемых неравенств для неизвестной c имеется ровно одно неизвестное неравенство, прием оно ограничивает c с противоположной стороны, то прием блокируется. В этой ситуации применяются другие приемы (см. ниже), обеспечивающие большее ускорение вычислений. Так как преобразуемые неравенства возникают в задаче не одновременно, а инициализация приема происходит по какому-то одному из них, предусмотрены несколько уровней срабатывания - 1,3,5 и 7. Если на момент появления инициализирующего неравенства второе неравенство отсутствовало, но появилось не позднее уровня 7, то прием все-таки сработает. Для задач с целью "пример" и задач, имеющих более одной целочисленной переменной, срабатывание на уровне 1 блокируется. Заменяющее утверждение обрабатывается множеством нормализаторов, причем собственно неравенства с c никак не нормализуются (они уже имелись в задаче). Неравенства с разностью a и b преобразуются нормализаторами неравенств для известных параметров "стандменьше", "стандменьшеилиравно". По мере возможности, эти неравенства явно разрешаются относительно параметров. Каждая из конъюнкций неравенств и вся дизъюнкция обрабатываются нормализатором "склейканеравенств". Наконец, все заменяющее утверждение обрабатывается нормализатором "нормлог". В результате, как правило, дизъюнкция из заменяющего терма исключается, и остается лишь там, где сравнение выражений a, b неочевидно. Во всех случаях заменяющий терм снабжается комментарием "разборслучаев" (при устраненной дизъюнкции он далее игнорируется).

Если остаются всего три неизвестных неравенства, два из которых ограничивают неизвестную с одной стороны, а третье - с другой, то применяются следующие приемы, выполняющие с помощью оператора "склейканеравенств" попытку одновременного упрощения всей тройки неравенств:

$$\forall_{abcd}(a < c \,\&\, b < c \,\&\, c < d \leftrightarrow (0 \leq a-b \,\&\, a < c \vee \neg(0 \leq a-b) \,\&\, b < c) \,\&\, c < d)$$

$$\forall_{abcd}(c < a \,\&\, c < b \,\&\, d < c \leftrightarrow (0 \leq a-b \,\&\, c < b \vee \neg(0 \leq a-b) \,\&\, c < a) \,\&\, d < c)$$

$$\forall_{abcd}(c \leq a \,\&\, c < b \,\&\, d < c \leftrightarrow (a-b < 0 \,\&\, c \leq a \vee \neg(a-b < 0) \,\&\, c < b) \,\&\, d < c)$$

$$\forall_{abcd}(a \leq c \,\&\, b < c \,\&\, c < d \leftrightarrow (0 < a-b \,\&\, a \leq c \vee \neg(0 \leq a-b) \,\&\, b < c) \,\&\, c < d)$$

Приемы снабжены указателями "альтернатива", позволяющими обрабатывать также нестрогие встречные неравенства. Переменная c идентифицируется с неизвестной, выражения a, b, d известны. Задача не имеет других неизвестных неравенств и не имеет уравнений, пересекающихся с рассматриваемыми неравенствами по своим переменным. Неравенства не используются для сопровождения по о.д.з. Уровни срабатывания равны 1,3,5,7; заменяющий терм сопровождается комментарием "разборслучаев". Нормализация аналогична предыдущим приемам, но вместо внутренней дизъюнкции нормализуется весь заменяющий терм.

Если задача имеет цель "пример", то объединение односторонних неравенств выполняется с помощью операций "минимум" и "максимум":

$$\forall_{abx}(a < x \;\&\; b < x \leftrightarrow \max(a, b) < x)$$

$$\forall_{abx}(x < a \;\&\; x < b \leftrightarrow x < \min(a, b))$$

Оба неравенства суть условия задачи на описание. Переменная x идентифицируется с неизвестной, выражения a, b известны. Уровень срабатывания равен 2.

8. Учет невырожденности промежутка для неизвестной. На этапе редактирования ответа могут встретиться два противоположных неравенства для неизвестной x. Чтобы в ответе явно было обозначено условие невырожденности промежутка изменения x, добавляются соответствующие неравенства для концов этого промежутка:

   $$\forall_{abcd}(d = (b < c) \;\&\; b < x \;\&\; x < c \rightarrow d)$$

   $$\forall_{abcd}(d = (b < c) \;\&\; b \leq x \;\&\; x < c \rightarrow d)$$

   Приемы имеют заголовок "выводусловия". Они применяются на этапе редактирования ответа к двум неравенствам, являющимся условиями задачи на описание. Переменная x - неизвестная, выражения b, c известны, причем хотя бы одно из них неконстантное. Антецедент находит результат d обработки неравенства для b, c нормализатором "стандменьше". Для ускорения из посылок исключаются при этом все неизвестные утверждения. Проверяется, что d либо является константой "ложь", либо неконстантное. Если задача имеет дизъюнктивное условие без неизвестных, содержащее неравенства, то прием блокируется. Он будет применен внутри подзадач, возникающих при разборе случаев по данной дизъюнкции. Комментарий (связпеременная x) блокирует повторное применение приема. Приемы вводят комментарий "стандменьше", блокирующий дальнейшие попытки упрощения неравенств для параметров. Указатель "альтернатива(фикс(3)меньшеилиравно)" у первого приема разрешает вместо строгого неравенства $x < c$ идентифицировать нестрогое.

9. Усмотрение противоречия из двух противоположных неравенств для неизвестного выражения.

   $\forall_{abc}(a \leq c \;\&\; 0 \leq a - b \rightarrow \neg(c < b))$

   $\forall_{abc}(c < a \;\&\; 0 \leq b - a \rightarrow \neg(b < c))$

   $\forall_{abc}(c \leq a \;\&\; 0 \leq b - a \rightarrow \neg(b < c))$

   Приемы имеют заголовок "второйтерм" и заменяют соответствующее неравенство для c на константу "ложь". Это неравенство входит в условие задачи на описание. Либо оно совпадает с данным условием, причем выражение c содержит неизвестные и отсутствует уравнение, имеющее с неравенством общие переменные, либо имеет место этап редактирования ответа, неравенство не содержит неизвестных, c - неконстантное выражение, а a, b - константные. Первый антецедент идентифицируется с утверждением из контекста преобразуемого неравенства, второй - обрабатывается проверочным оператором. Чтобы облегчить проверку, предпринимается попытка разложить $b - a$ на множители. Уровни срабатывания равны 1,3 и 6. Имеется сильный ограничитель трудоемкости. Прием блокируется при наличии комментария "стандменьше", а также при редактировании ответа задачи на описание, возникшего после разбора случаев, так как попытки применения его в указанных ситуациях избыточны. Заметим, что при наличии двух строгих неравенств ситуация становится симметричной, и отдельный прием с консеквентом $\neg(c < b)$ не обязателен.

10. Квадратные неравенства.

    $\forall_{abcdepqr}(d = b^2 + 4ac \;\&\; p = \sqrt{d} \;\&\; q = (-b - p)/2a \;\&\; r = (-b + p)/2a \rightarrow c < ae^2 + be \leftrightarrow a < 0 \;\&\; 0 \leq d \;\&\; (p < 0 \;\&\; q < e \;\&\; e < r \;\vee\; 0 \leq p \;\&\; r < e \;\&\; e < q) \;\vee\; 0 < a \;\&\; (d < 0 \;\vee\; 0 \leq d \;\&\; (p < 0 \;\&\; (e < r \;\vee\; q < e) \;\vee\; 0 \leq p \;\&\; (e < q \vee\; r < e))) \;\vee\; a = 0 \;\&\; c < be)$

    $\forall_{abcdepqr}(d = b^2 + 4ac \;\&\; p = \sqrt{d} \;\&\; q = (-b - p)/2a \;\&\; r = (-b + p)/2a \rightarrow ae^2 + be < c \leftrightarrow 0 < a \;\&\; 0 \leq d \;\&\; (0 \leq p \;\&\; q < e \;\&\; e < r \;\vee\; p < 0 \;\&\; r < e \;\&\; e < q) \;\vee\; a < 0 \;\&\; (d < 0 \;\vee\; 0 \leq d \;\&\; (0 \leq p \;\&\; (e < r \;\vee\; q < e) \;\vee\; p < 0 \;\&\; (e < q \vee\; r < e))) \;\vee\; a = 0 \;\&\; be < c)$

    Преобразуемое неравенство является условием задачи на описание. Выражение e содержит неизвестные, выражения a, b, c - не содержат. Задача не имеет уравнений, пересекающихся с неравенством по своим переменным, и не имеет дизъюнктивных условий с неизвестными. Первый антецедент вычисляет дискриминант d, обращаясь последовательно к нормализаторам "нормплюс", "стандплюс", "видумножение". Второй антецедент упрощает квадратный корень из дискриминанта с помощью нормализатора "нормстепень" и отбрасывает внешний знак модуля, если он есть. Третий и четвертый антецеденты вычисляют корни трехчлена: предпринимается попытка разложить на множители числители дробей, и далее используется нормализатор общей стандартизации "нормдробь". Прием блокируется, если усматривается, что дискриминант равен нулю либо отрицателен. Для этих случаев предусмотрены особые приемы:

$\forall_{abcde}(b^2 + 4ac = 0 \to c < ae^2 + be \leftrightarrow 0 < a \ \& \ \neg(2ae + b = 0) \ \vee \ a = 0 \ \& \ c < 0)$

$\forall_{abcde}(b^2 + 4ac = 0 \to ae^2 + be < c \leftrightarrow a < 0 \ \& \ \neg(2ae + b = 0) \ \vee \ a = 0 \ \& \ 0 < c)$

$\forall_{abcde}(d = b^2 + 4ac \ \& \ d < 0 \to ae^2 + be < c \leftrightarrow a < 0)$

$\forall_{abcde}(d = b^2 + 4ac \ \& \ d < 0 \to c < ae^2 + be \leftrightarrow 0 < a)$

Уровни срабатывания всех перечисленных приемов равны 2.

11. Попытка раскрывания скобок с неизвестными слагаемыми.

$\forall_{abcdefg}(f = a(b + c)^d + e \to a(b + c)^d + e < g \leftrightarrow f < g)$

Неравенство является условием задачи на описание. Показатель степени d - натуральная константа, меньшая 6 (возможно, равная 1). Основание степени $b + c$ содержит неизвестные. Правая часть неравенства неизвестных не содержит. Либо a, либо d отлично от единицы; либо e, либо g отлично от 0. Задача не имеет уравнений, пересекающихся с преобразуемым неравенством по своим переменным. Выражение e не имеет дробных слагаемых, выражение a не содержит конечных сумм и произведений. Антецедент обращается к нормализаторам "нормплюс", "стандплюс", "уравнплюс". Заменяющее неравенство обрабатывается нормализатором "уравнменьше". Проверяется, что преобразование не увеличивает максимальной глубины вхождений ни одной из неизвестных, причем хотя бы для одной неизвестной - строго уменьшает такую глубину. Под глубиной вхождения понимается число его надтермов. В большинстве случаев это просто означает, что после раскрывания скобок нормализатор "уравнменьше" смог явно разрешить неравенство относительно неизвестных. Указатель "дробь(фикс(0 1)фикс(0 2))" разрешает применять прием с перестановкой частей неравенства. Уровень срабатывания равен 2.

12. Попытка разложения на множители разности частей неравенства.

$\forall_{abc}(c = b - a \to a < b \leftrightarrow 0 < c)$

Неравенство входит либо в условие задачи на описание, либо в посылку задачи на исследование. Хотя бы одно из выражений a, b содержит неизвестные и отлично от переменной. Если одна из частей неравенства нулевая, то другая не имеет мультипликативного заголовка ("умножение", "степень", "дробь"). Отсутствуют уравнения, пересекающиеся с неравенством по своим переменным. Задача не имеет цели "известно". Антецедент обращается к нормализаторам "нормплюс", "видумножение", "уравнплюс". Для оценки целесообразности преобразования применяется тот же оператор фильтра "фильтрмножителей", который возникал ранее в приеме разложения на множители разности частей уравнения. Необходимо, чтобы либо этот оператор оказался истинным, либо хотя бы одна из частей неравенства имела своим слагаемым неизвестную дробь, либо выражение c имело длину, меньшую чем хотя бы одно из выражений a, b. В последнем случае проверяется также, что ни одна из частей неравенства не является известной дробью. Если одна из частей неравенства - переменная, не входящая в другую часть и связанная внешним описателем "класс", то прием блокируется. При срабатывании приема происходит удаление комментариев, блокирующих повторную попытку перехода к новым неизвестным. Уровни срабатывания равны 3 и 6.

13. Возведение в квадрат неравенств с радикалами. Если число радикалов менее трех, то используются следующие приемы:

    $\forall_{abcde}(0 \leq c \;\&\; e = a^2c - (d-b)^2 \rightarrow a\sqrt{c} + d < b \leftrightarrow d < b \;\&\; (a < 0 \;\vee\; e < 0) \;\vee\; a < 0 \;\&\; 0 < e)$

    $\forall_{abcde}(0 \leq c \;\&\; e = a^2\sqrt{c} - (d-b)^2 \rightarrow a\sqrt[4]{c} + d < b \leftrightarrow d < b \;\&\; (a < 0 \;\vee\; e < 0) \;\vee\; a < 0 \;\&\; 0 < e)$

    Преобразуемое неравенство является условием задачи на описание. Выражение c содержит неизвестные, b - не содержит. Число слагаемых выражения d, имеющих своим сомножителем неизвестный радикал, не более одного. В первом приеме это выражение не имеет слагаемых, сомножителями которых являются неизвестные радикалы степени отличной от 2. В обоих приемах, если выбор выражения c неоднозначен, то берется самое длинное. Левая часть неравенства не должна иметь дробных слагаемых; задача не имеет уравнений, пересекающихся с преобразуемым неравенством по своим переменным. Указатели "дробь(...)" обеспечивают применение приема к неравенствам с радикалом в правой части. Первый антецедент обрабатывается проверочным оператором, второй - выполняет возведение неравенства в квадрат с помощью нормализаторов "нормплюс", "стандплюс", "уравнплюс". Уровень срабатывания равен 5.

    В случае трех радикалов создан следующий прием:

    $\forall_{abcdefghei}(i = (a\sqrt{f} + b\sqrt{g})^2 - (e - d - c\sqrt{h})^2 \rightarrow a\sqrt{f} + b\sqrt{g} + c\sqrt{h} + d < e \leftrightarrow c\sqrt{h} + d < e \;\&\; (a\sqrt{f} + b\sqrt{g} \leq 0 \;\vee\; i < 0) \;\vee\; a\sqrt{f} + b\sqrt{g} \leq 0 \;\&\; 0 < i)$

    Неравенство является условием задачи на описание. Выражения f, g, h содержат неизвестные, причем h - самое длинное из них. Остаточная сумма d не имеет слагаемых, сомножителями которых служили бы неизвестные радикалы. Задача не имеет уравнений, пересекающихся с неравенством по своим переменным. Как и выше, допутима перестановка частей неравенства, а уровень срабатывания равен 5. Наконец, возведение в квадрат неравенства с четырьмя радикалами предпринимается лишь при выполнении специального условия:

    $\forall_{abcdefghei}(a^2f + b^2g - c^2h - d^2i = 0 \;\&\; e = ab\sqrt{f}\sqrt{g} - cd\sqrt{h}\sqrt{i} \rightarrow a\sqrt{f} + b\sqrt{g} + c\sqrt{h} + d\sqrt{i} < 0 \leftrightarrow c\sqrt{h} + d\sqrt{i} < 0 \;\&\; (a\sqrt{f} + b\sqrt{g} \leq 0 \vee e < 0) \vee a\sqrt{f} + b\sqrt{g} \leq 0 \;\&\; 0 < e)$

    Уровень срабатывания здесь равен 4.

14. Возведение в степень двучленных неравенств.

    $\forall_{abcdn}(n - \text{натуральное} \;\&\; \neg(n - \text{even}) \rightarrow ab^{c/n} + d < 0 \leftrightarrow a^nb^c + d^n < 0)$

    Неравенство является условием задачи на описание. Выражения b, d содержат неизвестные. Антецеденты обрабатываются проверочными операторами. Указатель "дробь(...)" разрешает перестановку частей неравенства. Заметим, что c, n могут идентифицироваться с произвольными выражениями, в том числе неконстантными. Уровень срабатывания равен 2.

$\forall_{abcde}(e - \text{rational} \ \& \ \neg(\text{числитель}(e) - \text{even}) \ \& \ \neg(\text{знаменатель}(e) - \text{even}) \rightarrow ab^e + cd^e < 0 \leftrightarrow a^{1/e}b + c^{1/e}d < 0)$

Неравенство является условием задачи на описание либо посылкой задачи на исследование. Коэффициенты a, c идентифицируются как произведения всех известных сомножителей. Указатель "общаястепень(общаястепень x5 степень(x2 x5)степень(x4 x5))" обеспечивает выделение общего невырожденного показателя степени e в остальных сомножителях. Предварительно проверяется, что каждый неизвестный сомножитель слагаемого левой части неравенства имеет вид степени с неизвестным основанием. При редактировании ответа прием блокируется. Разрешается перестановка частей неравенства. Уровень срабатывания равен 1.

$\forall_{abcde}(e - \text{rational} \ \& \ \text{числитель}(e) - \text{even} \rightarrow ab^e + cd^e < 0 \leftrightarrow a < 0 \ \& \ c < 0 \ \& \ (\neg(b = 0) \ \vee \ \neg(d = 0)) \ \vee \ ac < 0 \ \& \ (|a|^{1/e}|b| - |c|^{1/e}|d|)\text{sg}a < 0)$

Прием аналогичен предыдущему, но применяется только к условиям задач на описание.

15. Редактирование ответа. При редактировании ответа задачи на описание предпринимается попытка упрощения сопровождающих неравенств с известными параметрами. По мере возможности, они явно разрешаются относительно параметров. Так как априори неясно, возможно ли добиться такого разрешения и нужно ли оно, используются лишь ослабленные средства, сгруппированные в пакетном нормализаторе "стандменьше". Обращение к нему выполняется следующим приемом:

    $\forall_{abc}(c = (a < b) \rightarrow a < b \leftrightarrow c)$

    Неравенство не содержит неизвестных и является условием задачи на описание, причем имеет место этап редактирования ответа. Антецедент обращается к нормализатору "стандменьше"; для ускорения вычислений из посылок исключаются все утверждения с неизвестными. Одновременно добавляются посылки, необходимые для сопровождения неравенства по о.д.з. Это нужно, так как старое сопровождение по о.д.з. при обработке ответа обычно портится из-за упрощения и явного разрешения сопровождающих утверждений. Задача не должна иметь комментария "стандменьше", указывающего на завершающую фазу редактирования. Созданы две версии приема. Первая из них применяется, если результат c не содержит связки "или", вторая - если неравенство не используется для сопровождения по о.д.з. Уровни срабатывания равны 1.

    При редактировании ответа предпринимается попытка усмотрения истинности утверждений вида $\neg(x = b)$, где x - неизвестная. Основанием данной попытки служит неравенство для x:

    $\forall_{abx}((0 \leq a - b) = \text{истина} \ \& \ a < x \rightarrow \neg(x = b))$

    $\forall_{abx}((0 \leq b - a) = \text{истина} \ \& \ x < a \rightarrow \neg(x = b))$

    Отрицание равенства расположено в условии задачи на описание, для которой предпринимается редактирование ответа. Второй антецедент представляет собой неравенство из контекста, первый - выделен указателем "идентификатор". Заметим, что обращение к проверочному оператору для установления истинности неравенства в рассматриваемой ситуации малоэффективно, ибо сопровождающие утверждения уже испорчены. Поэтому неравенство для разности a и

b обрабатывается нормализатором "стандменьшеилиравно", а результат сравнивается с константной "истина". Приемы срабатывают на уровне 4.

Простейшие приемы преобразования неравенств для параметров, разрешающие неконстантный фрагмент относительно констант, вынесены в сканирование задачи:

$\forall_{abc}(b < a + c \leftrightarrow b - c < a)$

$\forall_{abc}(a < b - c \leftrightarrow a + c < b)$

Первый прием выполняет замену слева направо, второй - справа налево. c группируется из всех константных слагаемых; b - константа. Остаток a невырожденный. Неравенство входит в редактируемый ответ и не содержит неизвестных. Уровни срабатывания равны 1 и 5, причем в первом случае задача должна иметь комментарий "стандменьше". Во втором случае такой комментарий вводится.

$\forall ab(a < -b \leftrightarrow b < -a)$

$\forall_{ab}(-b < a \leftrightarrow -a < b)$

Оба приема применяются слева направо. a - константа, выражение b неконстантное. Уровни срабатывания те же.

$\forall_{abc}(0 < b \rightarrow ab < c \leftrightarrow a < c/b)$

$\forall_{abc}(b < 0 \rightarrow ab < c \leftrightarrow c/b < a)$

Выражение b группируется из всех константных множителей, остаток a невырожденный. c - константа.

В заключение приведем два приема, выполняющих потенцирование логарифмического неравенства для параметров.

$\forall_{abcdex}(a < 0 \;\&\; b = c/a \;\&\; e = 1/d^{1/b} \rightarrow 0 < a\log_x d + c \leftrightarrow b = 0 \;\&\; (d-1 < 0 \;\&\; 0 < x-1 \;\vee\; 0 < d-1 \;\&\; x-1 < 0) \;\vee\; b < 0 \;\&\; (d-1 < 0 \;\&\; (0 < x-1 \;\vee\; x-e < 0) \;\vee\; 0 < d-1 \;\&\; (0 < x-e \;\vee\; x-1 < 0) \;\vee\; d-1 = 0) \;\vee\; 0 < b \;\&\; (d-1 < 0 \;\&\; 0 < x-1 \;\&\; x-e < 0 \;\vee\; 0 < d-1 \;\&\; 0 < x-e \;\&\; x-1 < 0))$

Неравенство входит в редактируемый ответ задачи на описание и не содержит неизвестных. a, c, d идентифицируются с константами, выражение x неконстантное. Указатель "дробь(...)" обеспечивает возможность перестановки частей неравенства. Уровень срабатывания равен 2.

$\forall_{abcde}(0 < b \;\&\; \neg(b-1 = 0) \rightarrow a\log_b c + d < e \leftrightarrow 0 < b-1 \;\&\; 0 < b^e - c^a b^d \;\vee\; b-1 < 0 \;\&\; b^e - c^a b^d < 0)$

Разрешаются только такие неконстантные слагаемые выражения d, которые либо имеют вид $k\log_b m$, где k константа, либо имеют неконстантный сомножитель, на который делится слагаемое показателя некоторой степени, встречающейся в неравенстве. Либо выражение a константное, либо неравенство имеет степень с неконстантным показателем, не расположенную внутри выражения $\log_b c$. Либо b, либо c неконстантное. Указатель "дробь(...)" разрешает перестановку частей неравенства. Уровень срабатывания равен 3.

16. Попытка доказать либо опровергнуть неравенство с неизвестными. Если неравенство не удается решить стандартными методами, то на достаточно высоком

уровне предпринимаются попытки доказать его либо противоположное неравенство:

$\forall_{ab}(0 < b - a \to a < b)$

$\forall_{ab}(0 \leq a - b \to \neg(a < b))$

Приемы имеют заголовок "второйтерм". Неравенство $a < b$ является условием задачи на описание и содержит неизвестные. Если одна часть неравенства - неизвестная, а другая не содержит неизвестных, то прием блокируется. Антецедент обрабатывается вспомогательной задачей на доказательство, имеющей максимальный уровень 5. Заметим, что попытки использования производной здесь не применяются, так как соответствующие приемы срабатывают на уровне 6. Приемы имеют средний ограничитель трудоемкости. Их уровень срабатывания равен 8.

17. Одновременное решение двух квадратных неравенств. Пусть даны два квадратных неравенства $c < ax^2 + bx, fx^2 + gx < h$, причем $0 < a, 0 < f$. Преобразуем их к виду $0 < afx^2 + bfx - cf, afx^2 + agx - ah < 0$. Если $j = bf - ag \neq 0$, то рассматриваемые параболы пересекаются в единственной точке $x = cf - ah/j$. В случае, когда эта точка лежит ниже оси ординат, решение неравенств образовано интервалом между корнями первого и второго трехчлена. С учетом взаимного расположения вершин парабол, определяемого знаком выражения j, получаем следующие два приема:

    $\forall_{abcdefghijk}(j = bf - ag \; \& \; j < 0 \; \& \; k = (cf - ah)/j \; \& \; (ak^2 + bk - c < 0) = \text{истина} \; \& \; 0 < a \; \& \; 0 < f \to c < ai^2 + bi \; \& \; fi^2 + gi < h \leftrightarrow (-g - \sqrt{g^2 + 4fh})/2f < i \; \& \; i < (-b - \sqrt{b^2 + 4ac})/2a)$

    $\forall_{abcdefghijk}(j = bf - ag \; \& \; 0 < j \; \& \; k = (cf - ah)/j \; \& \; (ak^2 + bk - c < 0) = \text{истина} \; \& \; 0 < a \; \& \; 0 < f \to c < ai^2 + bi \; \& \; fi^2 + gi < h \leftrightarrow (-b + \sqrt{b^2 + 4ac})/2a < i \; \& \; i < (-g + \sqrt{g^2 + 4fh})/2f)$

    Приемы имеют заголовок "заменусловия(второйтерм)". Они применяются к неравенствам, являющимся условиями задачи на описание. Выражение i содержит неизвестные, коэффициенты a, b, c, f, g, h - не содержат. Задача не имеет других уравнений либо неравенств с неизвестными. Хотя бы один из коэффициентов неконстантный. Первый, третий и четвертый антецеденты выделены указателем "идентификатор". Выражения j, $cf - ah$, $ak^2 + bk - c$ обрабатываются нормализатором "видумножение". Неравенство $ak^2 + bk - c < 0$ обрабатывается нормализатором "стандменьше". Второй антецедент проверяется с помощью вспомогательной задачи на доказательство, решаемой до уровня 4. Выводятся необходимые для сопровождения по о.д.з. условия положительности выражений под радикалами. Указатели "альтернатива" расширяют прием на случаи нестрогих неравенств. Так как приемы введены для ускорения вычислений, их применение должно опережать приемы раздельного решения неравенств. Поэтому уровень срабатывания выбран равным 1.

18. Потенцирование логарифмических неравенств.

    $\forall_{abcde}(0 < b \; \& \; \neg(b - 1 = 0) \to a \log_b c + d < e \leftrightarrow 1 < b \; \& \; c^a b^d < b^e \; \vee \; b < 1 \; \& \; b^e < c^a b^d)$

    Неравенство является условием задачи на описание, не имеющей пересекающихся с ним по своим переменным уравнений. Оно не используется для сопровождения по о.д.з. Хотя бы одно из выражений b, c содержит неизвестные. Либо выражение a известно, либо неравенство имеет степень с неизвестным показателем, не расположенную внутри выражения $\log_b c$. Разрешаются только такие неизвестные слагаемые остаточной суммы d, которые либо имеют вид $k \log_b A$, где k известно, либо имеют неизвестный сомножитель, на который делится слагаемое показателя какой-либо содержащейся в неравенстве степени. Показательные неравенства в заменяющем терме обрабатываются нормализатором "уравнменьше", предпринимающим попытку их явного разрешения. Заменяющая дизъюнкция и ее дизъюнктивные члены обрабатываются нормализатором "склейканеравенств". Преобразованное неравенство снабжается комментарием "общаястепень", блокирующим его логарифмирование. Указатель "дробь(...)" предусматривает возможность перестановки частей неравенства. Уровень срабатывания равен 2.

19. Логарифмирование показательных неравенств.

$\forall_{abcde}(0 < b \ \& \ e = d - \log_b(-c/a) \rightarrow 0 < ab^d + c \leftrightarrow 0 < a \ \& \ 0 \leq c \ \vee \ b - 1 = 0 \ \& \ 0 < a + c \ \vee \ 0 < a \ \& \ c < 0 \ \& \ (0 < b - 1 \ \& \ 0 < e \ \vee \ b - 1 < 0 \ \& \ e < 0) \ \vee \ a < 0 \ \& \ 0 < c \ \& \ (0 < b - 1 \ \& \ e < 0 \ \vee \ b - 1 < 0 \ \& \ 0 < e) \ \vee \ a = 0 \ \& \ 0 < c)$

Неравенство является условием задачи на описание, не используеым для сопровождения по о.д.з. Отсутствуют уравнения, пересекающиеся с данным неравенством по своим переменным. Показатель степени d содержит неизвестные. Выражение c не является суммой. Заметим, что известные слагаемые, если бы они были, оказались бы перенесены другими приемами в противоположную часть неравенства, Поэтому c обычно будет неизвестным. Проверяется отсутствие комментария к неравенству "общаястепень" (см. предыдущий прием). Уровни срабатывания равны 2 и 4. Если a либо c содержат степень с неизвестным показателем, не являющуюся обобщенным множителем этих выражений, то уровень срабатывания равен 4, иначе он равен 2. Кроме того, при срабатывании на уровне 2 проверяется отсутствие в неравенстве целочисленных неизвестных. Указатель "дробь(...)" обеспечивает учет возможной перестановки частей неравенства. При обработке заменяющего терма и во втором антецеденте используются нормализаторы общей стандартизации. Если обе части неравенства ненулевые, то применяются следующие приемы:

$\forall_{bcde}(0 < b \rightarrow d < eb^c \leftrightarrow (\neg(b = 1) \ \& \ d \leq 0 \ \vee \ (b < 1 \ \& \ c < -\log_b e + \log_b d \ \vee \ -\log_b e + \log_b d < c \ \& \ 1 < b) \ \& \ 0 < d) \ \& \ 0 < e \ \vee \ (\neg(b = 1) \ \& \ 0 \leq e \ \vee \ (b < 1 \ \& \ -\log_b(-e) + \log_b(-d) < c \ \vee \ c < -\log_b(-e) + \log_b(-d) \ \& \ 1 < b) \ \& \ e < 0) \ \& \ d < 0 \ \vee \ b = 1 \ \& \ 0 < e - d)$

$\forall_{bcde}(0 < b \rightarrow db^c < e \leftrightarrow (\neg(b = 1) \ \& \ d \leq 0 \ \vee \ (b < 1 \ \& \ -\log_b d + \log_b e < c \ \vee \ c < -\log_b d + \log_b e \ \& \ 1 < b) \ \& \ 0 < d) \ \& \ 0 < e \ \vee \ (\neg(b = 1) \ \& \ 0 \leq e \ \vee \ (b < 1 \ \& \ c < -\log_b(-d) + \log_b(-e) \ \vee \ -\log_b(-d) + \log_b(-e) < c \ \& \ 1 < b) \ \& \ e < 0) \ \& \ d < 0 \ \vee \ b = 1 \ \& \ 0 < e - d)$

Показатель степени c содержит неизвестные, противоположная часть неравенства известна. Если коэффициент при степени либо основание степени содержат неизвестные, то комментарий "общаястепень" к данному неравенству блокирует его логарифмирование. Как и выше, прием срабатывает на уровнях 2 и 4, с теми же ограничениями. Если основание степени не известно, причем неравенство уже содержит какой-либо логарифм, то вместо логарифмирования по основанию b предпринимается логарифмирование по "старому" основанию:

$\forall_{abcdef}(0<b \,\&\, e=d\log_f b-\log_f(c/a)\to c<ab^d \leftrightarrow 0<a \,\&\, c\le 0 \ \vee\ 0<a \,\&\, 0<c \,\&\, (0<f-1 \,\&\, 0<e \ \vee\ f-1<0 \,\&\, e<0) \ \vee\ a<0 \,\&\, c<0 \,\&\, (0<f-1 \,\&\, e<0 \ \vee\ f-1<0 \,\&\, 0<e) \ \vee\ a=0 \,\&\, c<0)$

Здесь b, d не известны, c известно. f - основание некоторого логарифма, уже имеющегося в неравенстве. Указатель "дробь(...)" учитывает возможность перестановки левой и правой частей. Чтобы прием применялся раньше приемов, логарифмирующих по основанию b, его уровень срабатывания выбран равным 1. В заключение приведем прием, ориентированный на простой частный случай:

$\forall_{abc}(0<a\to 0<a^b-a^c \leftrightarrow 0<1-a \,\&\, 0<c-b \ \vee\ 1-a<0 \,\&\, c-b<0)$

Уровень срабатывания здесь равен 1; никаких ограничений на зависимость a, b, c от неизвестных не накладывается. Допускается перестановка частей неравенства.

20. Неравенство третьей степени: случай отрицательного дискриминанта. Используется формула корней кубического четырехчлена для того случая, когда вещественный корень единственный.

    $\forall_{abcdpqrx}(\neg(a=0) \,\&\, p=(3ac-b^2)/3a^2 \,\&\, q=(2b^3-9abc-27a^2d)/27a^3 \,\&\, r=q^2/4+p^3/27 \,\&\, 0<r \,\&\, s=\sqrt[3]{-q/2+\sqrt{r}}+\sqrt[3]{-q/2-\sqrt{r}}-b/3a \to ax^3+bx^2+cx<d \leftrightarrow a<0 \,\&\, s<x \ \vee\ 0<a \,\&\, x<s)$

    Неравенство входит в условие задачи на описание. Выражение x содержит неизвестные, выражения a, b, c, d константные. Допускается перестановка частей. Уровень срабатывания равен 7. Антецеденты последовательно вычисляют промежуточные выражения p, q, r, s, используя нормализаторы общей стандартизации.

21. Существование решения неравенств. Если возникают неравенства для несущественных неизвестных задачи, то в ряде случаев удается от них избавиться, перейдя к эквивалентному условию на оставшиеся переменные. Прежде всего, рассмотрим группу приемов, относящихся к двум неравенствам, определяющим промежуток значений несущественной неизвестной:

    $\forall_{abc}(\exists_c(a<c \,\&\, c<b \,\&\, c - \text{число}) \leftrightarrow a<b)$

    $\forall_{ab}(\exists_c(0<a+c \,\&\, 0<b-c \,\&\, c - \text{число}) \leftrightarrow 0<a+b)$

    $\forall_{abde}(\exists_c(a<b+c \,\&\, d+c<e \,\&\, c - \text{число}) \leftrightarrow a+d<b+e)$

    $\forall_{abc}(0<a\to \exists_x(0<x \,\&\, ax+b<c \,\&\, x - \text{число}) \leftrightarrow b<c)$

    Все эти приемы имеют заголовок "связка". Переменная под квантором существования идентифицируется с несущественной неизвестной задачи на описание, подкванторные утверждения - с условиями задачи, причем все оставшиеся условия не содержат данной неизвестной. Прием исключает неизвестную, заменяя все содержащие ее условия на указанное в теореме неравенство. Фильтры приемов специально оговаривают, что прочие переменные приема идентифицируются с выражениями, не содержащими исключаемой неизвестной. Указатели "альтернатива" разрешают идентификацию одного из неравенств как нестрогого. Первый и четвертый приемы срабатывают на уровне 0, второй и третий - на уровне 2. В последнем приеме указатель "группировка(x1)" разрешает наличие нескольких слагаемых с x, группируемых под общий коэффициент a. Еще один прием с заголовком "связка" усматривает невырожденность промежутка для неизвестной:

$\forall_{ab}(0 < b \to \exists_x(0 < x \ \& \ x - \text{число} \ \& \ ax < b)$

Уровень срабатывания здесь равен 0.

Если в задаче упоминается точная верхняя либо нижняя грань некоторого непустого ограниченного множества и имеется строгое неравенство для несущественной неизвестной из этого множества, то можно перейти к эквивалентному неравенству для соответствующей точной грани. Такой переход не всегда целесообразен, и вместо "необратимой" замены условий прием предпринимает лишь попытку замены, обращаясь к вспомогательной задаче. При неудаче решение будет продолжено с текущей ситуации. Созданы два приема:

$\forall_{abx}(0 \leq b \ \& \ \text{огрсверху}(a) \ \& \ \neg(a = \emptyset) \ \& \ (y \in a) = P(y) \ \& \ x < b\sup a \to \exists_y(P(y) \ \& \ x < by))$

$\forall_{abx}(0 \leq b \ \& \ \text{огрснизу}(a) \ \& \ \neg(a = \emptyset) \ \& \ (y \in a) = P(y) \ \& \ b\inf a < x \to \exists_y(P(y) \ \& \ by < x))$

Приемы имеют заголовок "существует". y - несущественная неизвестная задачи на описание, встречающаяся в неравенстве $x < by$, где x, b от y не зависят. Идентификация начинается с этого неравенства. Затем указатель "контекст(посылка(x3)позиция(x4 x3)вид(x4 суп(x1)))" (либо аналогичный указатель с "инф") обеспечивает усмотрение посылки, содержащей упоминание о точной грани. Так идентифицируется множество a. Первые три антецедента обрабатываются проверочными операторами и отсекают непригодные a. Четвертый антецедент выделен указателем "идентификатор". Он обрабатывает условие принадлежности неизвестной y множеству a нормализаторами "норм", "нормпринадлежит". Таким образом идентифицируется утверждение $P(y)$. Проверяется, что оно является условием задачи и что других содержащих y условий (кроме рассматриваемого неравенства) нет. Пятый антецедент выделен указателем "подборзначений". На него будут заменяться условия с y во вспомогательной задаче. Уровень срабатывания равен 4.

Следующий прием исключает целочисленную либо рациональную несущественную неизвестную:

$\forall_{ab}(0 < a \ \& \ b - \text{число} \to \exists_n(n - \text{натуральное} \ \& \ 0 < an + b))$

Заголовок приема - "связка". Указатель "вариант(фикс(0 2 1)целое рациональное)" разрешает вместо условия "$n-$натуральное" рассматривать условия "$n-$ целое", "n − рациональное". Уровень срабатывания равен 0.

В заключение приведем еще два приема с заголовком "существует", обращающихся к вспомогательным задачам. Первый из них связан с условием существования рационального приближения:

$\forall_{abcdmnA}(0 < c \ \& \ \text{set}_x(A(x)) = \{m, \ \ldots \ , n\} \ \& \ |b| < cd \ \& \ -bm - cd < ac \ \& \ ac < cd - bn \to \exists_x(A(x) \ \& \ |a + bx/c| < d))$

Неравенство $|a + bx/c| < d$ является условием задачи на описание, имеющей цель "пример". x - целочисленная несущественная неизвестная, не входящая в a, b, c. Хотя бы одно из выражений b, c, d содержит неизвестные. $A(x)$ идентифицируется с конъюнкцией остальных содержащих x условий. Второй антецедент упрощает с помощью вспомогательной задачи выражение для класса удовлетворяющих $A(x)$ значений и идентифицирует его с конечным отрезком целых чисел $\{m, \ \ldots \ , n\}$. Третий, четвертый и пятый антецеденты замещают условия с x во вспомогательной задаче. Уровень срабатывания равен 3. Второй прием связан с исключением модуля:

$\forall_{pq}(0 < p(-q) \rightarrow \exists_x(x - \text{число} \ \& \ 0 < p(x) - |x + q|))$

Решаемая задача на описание имеет цель "пример". Антецедент замещает во вспомогательной задаче условия с неизвестной x. Уровень срабатывания приема равен 4.

22. Сведение однородного показательного неравенства к квадратному. Приемы преобразуют однородное показательное неравенство в квадратное неравенство относительно показательного выражения X, после чего обращаются к процедуре "уравнменьше", решающей это неравенство относительно X. Если квадратное неравенство непосредственно заносить в список условий, то приемы стандартизации вернут его в исходный вид прежде, чем сработает прием для решения квадратных неравенств.

    $\forall_{abcdefghijk}(0 < a \ \& \ b - 2c = f \ \& \ g - 2d = e \rightarrow hk^b + ik^c a^d + ja^g < 0 \leftrightarrow hk^f(k^c/a^d)^2 + i(k^c/a^d) < -ja^e)$

    Неравенство является условием задачи на описание либо посылкой задачи на исследование. Выражения b, c, d, g содержат неизвестные, выражения h, i, j - не содержат. Второй и третий антецеденты, выделенные указателем "идентификатор", упрощают разности показателей. Требуется, чтобы результаты e, f не содержали неизвестных. Если f не равно 0, то выражение k должно не содержать неизвестных. Аналогично, если e не равно 0, то выражение a должно быть известным. Указатель "дробь" обеспечивает применение приема при перестановке частей неравенства. Уровень срабатывания этого и последующих приемов равен 2.

    $\forall_{abcdefghijk}(0 < d \ \& \ k = b - c \ \& \ k - c + j = 0 \rightarrow ad^{b+i} + ed^{c+g} + fd^{h+j} < 0 \leftrightarrow ad^i(d^k)^2 = ed^g d^k + fd^h < 0)$

    Неравенство является условием задачи на описание. Выражения b, c, j содержат неизвестные, d - не содержит. Выражения g, h, i группируются из всех известных слагаемых показателей. Допускается перестановка частей неравенства.

    $\forall_{abcdefghijk}(0 < a \ \& \ b - 2c = f \ \& \ d = ae \ \& \ g - 2c = h \rightarrow ie^b + jd^c + ka^g < 0 \leftrightarrow ie^f((e/a)^c)^2 + j(e/a)^c < -ka^h)$

    Неравенство является условием задачи на описание либо посылкой задачи на исследование. Выражения b, c, g содержат неизвестные, f, i, j, k, h - не содержат. Антецеденты со второго по четвертый выделены указателем "идентификатор". Третий антецедент убеждается в том, что результат упрощения произведения оснований степени a, e равен основанию d.

23. Переход к условию принадлежности промежутку при подборе примера.

    $\forall_{abx}(x - \text{число} \ \& \ a < x \ \& \ x < b \leftrightarrow x \in (a, b))$

    Прием имеет заголовок "заменаусловия(второйтерм)". Неравенства являются условиями задачи на описание, имеющей цель "пример". x - неизвестная, выражения a, b известны. Проверяется отсутствие других условий, содержащих x. Уровень срабатывания равен 3.

24. Исключение внешнего квантора путем перехода к максимумам и минимумам. Приемы применяются к неравенствам, на которые навешен квантор общности:

    $\forall_{abcdfg}(d = \mathrm{Min}(\lambda_x(g(x), x - \text{число}), \mathrm{set}_x(f(x)), b, c) \to \forall_x(f(x) \to a < g(x)) \leftrightarrow \forall_{bc}(d \to a < c))$

    $\forall_{abcdfg}(d = \mathrm{Max}(\lambda_x(g(x), x - \text{число}), \mathrm{set}_x(f(x)), b, c) \to \forall_x(f(x) \to g(x) < a) \leftrightarrow \forall_{bc}(d \to c < a))$

    Рассмотрим лишь первый прием, так как второй аналогичен. Кванторная импликация $\forall_x(f(x) \to a < g(x))$ является условием задачи на описание. Выражение a содержит неизвестные, а утверждение $f(x)$ и выражение $g(x)$ - не содержат. Переменные f, g функциональные. Вводятся новые переменные b, c, и антецедент обращается к вспомогательной задаче на описание с условием $\mathrm{Min}(\lambda_x(g(x), x - \text{число}), \mathrm{set}_x(f(x)), b, c)$. Оно означает, что b есть множество точек, в которых функция $\lambda_x(g(x), x - \text{число})$ достигает минимальное значение c на множестве $\mathrm{set}_x(f(x))$. Неизвестные задачи - b, c. Если задачу удается решить, то ответ на нее идентифицируется с d. Это - некоторое утверждение с переменными b, c, относительно которых оно явно разрешено. Поэтому навешивание в заменяющем терме квантора общности по b, c имеет локальный характер; ближайшие срабатывания приемов этот квантор устранят. Уровень срабатывания приемов равен 2.

25. Использование синтезаторов "верхняяоценка", "нижняяоценка" для устранения зависимости неравенства от заданных переменных. Приемы применяются, если задача имеет цель (независит $x_1, \ldots, x_n$), перечисляющую неизвестные, которые не должны входить в ответ. В случае задачи на описание прием имеет заголовок "подборзначений":

    $\forall_{abc}(a < b \;\&\; b \leq c \to a < c)$

    Здесь $a < c$ - условие задачи, у которого левая часть a не содержит запрещенных (заданных целью (независит $\ldots$)) переменных, а правая - содержит. В задаче не требуется получить полное описание ,т.е. либо есть цель "пример", либо нет цели "полный". Второй антецедент, выделенный указателем "значения", обращается к синтезатору "нижняяоценка", получающему для выражения b оценку c. Этому синтезатору передается в качестве комментария цель (независит $\ldots$), и он может выдать лишь такой результат b, который не содержит запрещенных переменных. Первый антецедент выделен указателем "подборзначений". Прием предпринимает попытку обращения к вспомогательной задаче, полученной из текущей задачи заменой условия $a < c$ на этот антецедент. Уровень срабатывания равен 3.

    Для задач на исследование созданы аналогичные приемы, которые вместо обращения к вспомогательной задаче выводят новое неравенство, не содержащее запрещенных переменных. В случае верхней оценки теорема приема та же, что и выше. Однако, теперь $a < b$ - посылка задачи, причем b содержит запрещенные переменные, a - содержит неизвестные. Задача должна иметь цель "длялюбого", т.е. решается для исключения кванторной импликации в условиях внешней задачи на описание (см. приемы символа "длялюбого", реализованные на ЛОСе). После определения вторым антецедентом верхней оценки c для выражения b выводится следствие $a < c$, не имеющее в своей правой части запрещенных переменных. Чтобы исключать запрещенные переменные в левой части, создан прием следующего вида:

$\forall_{abc}(a < b \ \& \ c \leq a \rightarrow c < b)$

Уровни срабатывания приемов равны 3.

Доказательство неравенств

Основная часть элементарного аппарата, используемого для доказательства строгих неравенств, собрана в проверочных операторах "усмменьше", "провменьше" (второй из них реализует цепочки оценок с помощью неравенства для среднего арифметического и среднего геометрического). Аппарат, использующий пределы и производные, будет изложен в разделах, посвященных приемам решения задач по математическому анализу. Здесь же приведем те немногие остающиеся элементарные приемы, которые реализуются при непосредственном сканировании задачи.

1. Раскрывание скобок. Если ненулевая часть неравенства имеет слагаемое, допускающее раскрывание скобок, то это преобразование применяется ко всей части:

   $\forall_{abcdef}(f = a(b+c)^d + e \rightarrow a(b+c)^d + e < 0 \leftrightarrow f < 0)$

   $\forall_{abcdef}(f = a(b+c)^d + e \rightarrow 0 < a(b+c)^d + e \leftrightarrow 0 < f)$

   Неравенство является условием задачи на доказательство. Антецедент обращается к нормализаторам "нормплюс", "стандплюс". Если ненулевая часть имеет слагаемое с натуральной степенью суммы, показатель которой более 4, то прием блокируется. Уровень срабатывания равен 2.

2. Сложение дробей.

   $\forall_{abcd}(d = a/b + c \rightarrow 0 < a/b + c \leftrightarrow 0 < d)$

   Неравенство является условием задачи на доказательство, имеющей максимальный уровень не менее 5. Антецедент обращается к оператору "видумножение". Чтобы избежать трудоемкой попытки разложения на множители в сомнительных случаях (одновременное присутствие разнородных тригонометрических, логарифмических и показательных выражений), используется оператор фильтра "фильтрнеравенства". Он будет рассмотрен в разделах, посвященных доказательству неравенств с помощью производной, где введен для блокировки элементарных средств доказательства. Уровень срабатывания равен 1.

3. Возведение в квадрат неравенства с одним радикалом.

   $\forall_{abc}(0 \leq a \ \& \ b \leq 0 \ \& \ 0 \leq c \rightarrow 0 < a + b\sqrt{c} \leftrightarrow 0 < a^2 - b^2 c)$

   $\forall_{abc}(a \leq 0 \ \& \ 0 \leq b \ \& \ 0 \leq c \rightarrow 0 < a + b\sqrt{c} \leftrightarrow 0 < b^2 c - a^2)$

   Неравенство является условием задачи на доказательство. Для блокировки приема используется оператор "фильтррадикалов". Он проверяет выполнение следующих условий:

   (a) c есть сумма либо имеет сумму своим сомножителем.

   (b) Если текущий уровень равен 2, то либо a не является суммой, либо не имеет слагаемых с сомножителем - дробной степенью суммы.

   (c) Если текущий уровень равен 4, то a имеет не более двух слагаемых с сомножителем - дробной степенью суммы.

   (d) В a отсутствует сомножитель слагаемого, имеющий вид дробной степени выражения, боле длинного, чем c.

   В ненулевой части заменяющего терма выполняется раскрывание скобок. Указатель "дробь" разрешает перестановку частей неравенства. Уровни срабатывания равны 2 и 4.

4. Возведение в квадрат неравенства с двумя радикалами.

   $\forall_{abcde}(0 \leq b \; \& \; 0 \leq d \; \& \; 0 \leq a \; \& \; 0 \leq c \; \& \; e \leq 0 \rightarrow 0 < a\sqrt{b} + c\sqrt{d} + e \leftrightarrow 0 < (a\sqrt{b} + c\sqrt{d})^2 - e^2)$

   $\forall_{abcde}(0 \leq b \; \& \; 0 \leq d \; \& \; a \leq 0 \; \& \; c \leq 0 \; \& \; 0 \leq e \rightarrow 0 < a\sqrt{b} + c\sqrt{d} + e \leftrightarrow 0 < e^2 - (a\sqrt{b} + c\sqrt{d})^2)$

   Неравенство является условием задачи на доказательство. Хотя бы одно из выражений b, d является суммой либо имеет своим сомножителем сумму. Выражение e не имеет слагаемого, имеющего своим сомножителем степень с дробным показателем и неконстантным основанием. В заменяющем терме выполняется раскрывание скобок. Допускается перестановка частей. Уровень срабатывания равен 3.

5. Разность квадратов.

   $\forall_{ab}(0 \leq b \; \& \; 0 \leq a \rightarrow 0 < b^2 - a^2 \leftrightarrow 0 < b - a)$

   Неравенство является условием задачи на доказательство. Допускается перестановка частей. Уровень срабатывания равен 2.

6. Разложение на множители. Предпринимается попытка разложить на множители ненулевую часть неравенства:

   $\forall_{abc}(c = a + b \rightarrow 0 < a + b \leftrightarrow 0 < c)$

   Неравенство является условием задачи на доказательство. Антецедент обращается к оператору "видумножение". Проверяется, что результат имеет мультипликативный заголовок. Для отсечения малоперспективных попыток используется упоминавшийся выше оператор фильтра "фильтрнеравенства". Уровень срабатывания равен 5.

7. Вывод следствий в посылках. Обычно при доказательстве неравенств посылки берутся "готовыми" и не преобразуются. Однако, в нескольких задачах понадобились приемы, выводящие новые посылки:

   (a) Неотрицательность разности квадратов.
   $\forall_{ab}(0 < a^2 - b^2 \; \& \; 0 \leq a \rightarrow 0 < a - b \; \& \; 0 < a + b)$
   Уровень срабатывания равен 3.

   (b) Сложение неравенств.
   $\forall_{abc}(0 < a + b \; \& \; 0 \leq c - b \rightarrow 0 < a + c)$
   Исключаемое выражение b и результат $a + c$ не должны быть константными. Либо каждое неконстантное слагаемое выражения a должно иметь подобное (с точностью до константных коэффициентов) слагаемое в c, либо каждое неконстантное слагаемое выражения c должно иметь подобное слагаемое в a. Уровень срабатывания равен 2.

$\forall_{abcd}(0 < a + b - c \; \& \; 0 < d + c - b \to 0 < a + d)$

Выражения b, c неконстантные. Число слагаемых в результирующей сумме (после применения к ней нормализатора "нормплюс") не более 3. Уровень срабатывания равен 4.

(c) Вывод из неравенства для суммы квадратов.

$\forall_{abc}(0 < a^2 + b^2 - c \; \& \; 0 \leq a + b \; \& \; 0 \leq c \to 0 < a + b - \sqrt{c})$

Прием используется для "сближения" степени выражения в посылках со степенью этого же выражения в условии. Требуется, чтобы в условии встречалась сумма, некоторые слагаемые которой, при возведении их в натуральную степень (возможно, равную 1), давали бы a и b. Выражение c предполагается константным. Второй и третий антецеденты обрабатываются проверочными операторами. Уровень срабатывания равен 2.

$\forall_{abc}(0 < c - a^2 - b^2 \to 0 < \sqrt{2c} - a - b)$

Фильтры те же, что в предыдущем случае. Уровень срабатывания равен 1.

(d) Регистрация неравенства, полученного группировкой в одной части всех ненулевых членов.

$\forall_{ab}(b < a \to 0 < a - b)$

Одна часть неравенства является переменной, а другая ненулевой константой (в этом случае эквивалентное преобразование к виду неравенства с нулем в одной части будет заблокировано). Уровень срабатывания равен 0.

(e) Линейная комбинация неравенств для исключения переменной. Если доказываемое неравенство не содержит некоторой переменной, а неравенства в посылках содержат ее, то бывает полезно рассмотрение линейных комбинаций посылок, исключающих эту переменную. Аналогичные действия помогают выявить противоречивость посылок в задачах на исследование. Здесь создан следующий прием:

$\forall_{abcdx}(0 < a \; \& \; c < 0 \; \& \; 0 < ax + b \; \& \; 0 < cx + d \to 0 < ad - bc)$

Два последних антецедента идентифицируются с неравенствами из посылок задачи на доказательство либо задачи на исследование, имеющей цель "противоречие". Указатель "перегруппировка(фикс(3)фикс(4))" обеспечивает автоматическую группировку всех ненулевых слагаемых в требуемой части неравенства. Указатель "контекст(входит(x24 параметры(x23)))" определяет выбор некоторой переменной y, входящей в выражение x. Она не должна входить в выражения a, b, c, d. Если задача имеет тип "доказать", то y не входит в условие. Чтобы упорядочить процесс исключения переменных, используется комментарий к посылкам (группировки $y_1 \ldots y_n$), перечисляющий в порядке приоритетности исключаемые переменные $y_1, \ldots, y_n$. Если хоты бы одно из рассматриваемых двух неравенств содержит переменную y_i, то y должно быть переменной y_j при некотором $j < i$. Указатель "Примечание(группировки x24)" выполняет добавление к указанному списку новой переменной y. Уровень срабатывания равен 5. Фактически по теореме созданы два приема, отличающиеся указателями "альтернатива", разрешающими одному из неравенств быть нестрогим.

(f) Переход к новым основаниям степени в показательных неравенствах при доказательстве неравенства по индукции.

$\forall_{abcdepqm}(0 < a \,\&\, 0 < c \,\&\, 0 < c - a \,\&\, 0 < d \,\&\, e - a - m + c = 0 \,\&\, 0 < m - c \,\&\, 0 < pa^b + qc^d \,\&\, q < 0 \rightarrow 0 < pa^{b-d}e^d + qm^d)$

Указатель "контрольвывода(плюс(умножение(x17 степень(x5 x7)) умножение(x18 степень(x13 x8))))" инициирует применение приема при усмотрении в условии задачи на доказательство суммы, имеющей сомножителями двух своих слагаемых выражения e^g, m^h. Неравенство $0 < pa^b + qc^d$ (т.е. седьмой антецедент) идентифицируется с посылкой, помеченной как индуктивное предположение при доказательстве индукцией по некоторому параметру i. Проверяется, что этот параметр входит в показатели b, d, g, h и не входит в $b - d$. Оставшиеся антецеденты обрабатываются проверочными операторами. Прием использует легко вытекающую из них нижнюю оценку m для дроби ce/a. Выведенное неравенство помечается таким же комментарием, как исходное индуктивное предположение. Допускается рассмотрение нестрогих неравенств вместо строгих. Уровень срабатывания равен 5.

8. Усмотрение квадратного трехчлена. Прием усматривает положительность квадратного трехчлена с положительным старшим коэффициентом и отрицательным дискриминантом:

   $\forall_{abcx}(0 < c \,\&\, b^2 - 4ac < 0 \rightarrow 0 < ax^2 + bx + c)$

   Заголовок приема - "второйтерм". Неравенство входит в условие задачи на доказательство. Выражение x неконстантное. Коэффициенты a и b идентифицируются путем группировки всех слагаемых, делящихся на x^2 и x соответственно. Никакое слагаемое не делится на прочие степени выражения x. Антецеденты обрабатываются проверочными операторами, причем второй антецедент, выделенный указателем "проверка", обрабатывается усиленным проверочным оператором "провменьше". Уровень срабатывания равен 4.

9. Декомпозиция неравенства для суммы слагаемых одного знака. Если удается усмотреть, что каждое из двух слагаемых неотрицательно, то для доказательство строгой положительности суммы достаточно доказать отличие от нуля хотя бы одного слагаемого:

   $\forall_{ab}(0 \leq a \,\&\, 0 \leq b \rightarrow 0 < a + b \leftrightarrow \neg(a = 0) \;\vee\; \neg(b = 0))$

   Неравенство является условием задачи на доказательство. Для усмотрения неотрицательности используются проверочные операторы, причем введены сильные ограничители трудоемкости проверок. Уровень срабатывания равен 1.

10. Использование оператора "верхняяоценка". Для доказательства неравенства может быть предпринята попытка оценить сверху разность его частей и доказать отрицательность найденной оценки:

    $\forall_{abc}(a - b \leq c \,\&\, c < 0 \rightarrow a < b)$

    Неравенство является условием задачи на доказательство. Первый антецедент выделен указателем "значения". Он обращается к пакетному синтезатору "верхняяоценка". Проверяется, что найденная оценка константная. Второй антецедент обрабатывается проверочным оператором. Уровень срабатывания равен 5.

11. Использование степенной нижней оценки для части множителей факториала.

    $\forall_{abcmn}(0 \le m - n - b + 1 \;\&\; 0 \le a \;\&\; 0 \le c \;\&\; 0 \le b \;\&\; 0 < c(m-n)! - a \;\&\; n -$ натуральное & m – натуральное $\to 0 < cm! - ab^n)$

    Неравенство является условием задачи на доказательство. Выражение n неконстантное. Пятый антецедент обрабатывается вспомогательной задачей на доказательство, решаемой с тем же максимальным уровнем, что и текущая задача. Остальные антецеденты обрабатываются проверочными операторами. Уровень срабатывания равен 5.

12. Использование индуктивного предположения при доказательстве показательного неравенства.

    $\forall_{abcpqrs}(0 < a \;\&\; 0 < p \;\&\; 0 < r \;\&\; 0 < ra^c + s \;\&\; 0 \le qr - spa^{b-c} \to 0 < pa^b + q)$

    Неравенство входит в условие задачи на доказательство, имеющей комментарий (натуральное d). Этот комментарий определяет индуктивный параметр d. Четвертый антецедент представляет собой посылку задачи. Проверяется, что d входит как в b, так и в c. После этого предпринимается попытка установить вспомогательной задачей на доказательство истинность пятого антецедента, выводящего рассматриваемое неравенство из данной посылки. Допускается рассмотрение нестрогого неравенства. Уровень срабатывания равен 6.

13. Использование неравенства Бернулли.

    $\forall_{abcdn}(0 < n(a-d) - (b+d)(c-1) \;\&\; 0 < b+d \;\&\; n -$ натуральное & $0 < a + b \to 0 < (a+b)^n - c(b+d)^n)$

    Прием имеет заголовок "воройтерм". Неравенство входит в условие задачи на доказательство. Первый антецедент выделен указателем "легковидеть". Он обрабатывается вспомогательной задачей на доказательство, решаемой до максимального уровня 4. Остальные антецеденты обрабатываются проверочными операторами. Уровень обращения равен 3.

14. Использование степенной нижней оценки для части множителей факториала.

    $\forall_{abcmn}(0 \le m - n - b + 1 \;\&\; 0 \le a \;\&\; 0 \le c \;\&\; 0 \le b \;\&\; 0 < c(m-n)! - a \;\&\; n -$ натуральное & m – натуральное $\to 0 < cm! - ab^n)$

    Неравенство является условием задачи на доказательство. Выражение n неконстантное. Пятый антецедент обрабатывается вспомогательной задачей на доказательство, решаемой с тем же максимальным уровнем, что и текущая задача. Остальные антецеденты обрабатываются проверочными операторами. Уровень срабатывания равен 5.

Подбор примера

Если при решении задачи на описание, имеющей цель "пример", для неизвестной x остается единственное невырожденное условие, имеющее вид строгого неравенства, то используется прием, указывающий некоторое конкретное значение, при котором это неравенство выполнено:

$\forall_{ax}(x = a + 1 \to a < x)$

$\forall_{ax}(x = a - 1 \to x < a)$

$\forall_{abcx}(0 < b \ \& \ \neg(c = 0) \ \& \ x = -a/c \rightarrow |cx + a| < b)$

Приемы имеют заголовок "подборзначений". Неравенство в консеквенте является условием задачи на описание, имеющей цель "пример". Переменная x - неизвестная, остальные переменные идентифицируются с выражениями без неизвестных. Единственным отличным от неравенства условием, содержащим x, может быть утверждение "число(x)". В каждом приеме последний антецедент выделен указателем "подборзначений" - он будет перенесен во вспомогательную задачу. Остальные антецеденты обрабатываются проверочными операторами. Уровень срабатывания приема равен 1.

Неравенства с целочисленной неизвестной

1. Усиление неравенства с использованием условия целочисленности. В рассматриваемых ниже приемах предполагается, что неравенство входит в условие задачи на описание.

   $\forall_{abn}(n - \text{целое} \ \& \ b = [a] \rightarrow a < n \leftrightarrow b + 1 \leq n)$

   n идентифицируется с переменной (не обязательно неизвестной), a - с известным выражением. Отсутствуют уравнения, пересекающиеся с неравенством по своим переменным. Переменные неравенства не связаны внешними кванторами и описателями. Первый антецедент обрабатывается проверочным оператором, второй - преобразует выражение $[a]$ нормализатором "нормцелаячасть". Проверяется, что результат b не содержит символа "целаячасть" (исключение делается лишь для задач с целью "мощность", используемых в комбинаторике), а также символов $\sup, \inf$. Уровень срабатывания равен 2. Для противоположного неравенства создан прием с тем же уровнем срабатывания:

   $\forall_{abn}(n - \text{целое} \ \& \ b = [a] \ \& \ 0 < a - b \rightarrow n < a \leftrightarrow n \leq b)$

   Как видно из третьего антецедента, прием рассчитан лишь на случаи, когда целая часть оказывается строго меньше исходного выражения. Это ограничение снимается в следующем приеме:

   $\forall_{abn}(n - \text{целое} \ \& \ b = -[-a] \rightarrow n < a \leftrightarrow n \leq b - 1)$

   Здесь n идентифицируется с неизвестной, a - с известным выражением, причем не должно усматриваться, что a целочисленное. Уровень срабатывания приема равен 7. Для специальных случаев введен еще один вариант приема:

   $\forall_{ax}(x - \text{целое} \rightarrow x < a \leftrightarrow x \leq [a] \ \& \ \neg(a - \text{целое}) \ \vee \ a - \text{целое} \ \& \ x \leq a - 1)$

   Задача на описание должна иметь цель "пример" либо цель "истинность". x идентифицируется с выражением, содержащим неизвестные, a известно. Уровень срабатывания равен 4. В заключение приведем два приема, где целая часть берется от константного выражения:

   $\forall_{abcn}(c = [-a/b] \ \& \ 0 < a + bc \ \& \ b < 0 \ \& \ n - \text{целое} \rightarrow 0 < a + bn \leftrightarrow 0 \leq c - n)$

   $\forall_{abcn}(c = -[a/b] \ \& \ 0 < a + bc \ \& \ 0 < b \ \& \ n - \text{целое} \rightarrow 0 < a + bn \leftrightarrow 0 \leq n - c)$

   Выражения a, b - десятичные константы. Неравенство входит в условие задачи на описание. Для ускорения введен фильтр, проверяющий наличие в списке условий утверждения с заголовком "целое" либо "натуральное". Уровень срабатывания равен 1.

2. Переход к нестрогому неравенству. В подразделе рассматриваются приемы, аналогичные приемам предыдущего подраздела, но обе части неравенства уже целочисленные.

   $\forall_{xn}(n -$ целое $\&\ x -$ целое $\rightarrow x < n \leftrightarrow x \leq n - 1)$

   $\forall_{xn}(n -$ целое $\&\ x -$ целое $\rightarrow n < x \leftrightarrow n + 1 \leq x)$

   В этих приемах неравенство является условием задачи на описание, x - неизвестная, n известно. Уровень срабатывания равен 1. Для неравенств, переменная которых связана внешним описателем, созданы следующие два приема:

   $\forall_{xn}(n -$ целое $\&\ x -$ целое $\rightarrow x < n \leftrightarrow x \leq n - 1)$

   $\forall_{xn}(n -$ целое $\&\ x -$ целое $\rightarrow n < x \leftrightarrow n + 1 \leq x)$

   Переменная x входит в связывающую приставку внешнего описателя "класс", причем выражение n не зависит от переменных этой приставки. Уровень срабатывания равен 3. Еще один связанный с описателем "класс" прием выполняет стандартизацию вида условий на множество упорядоченных пар натуральных чисел $i, j, i < j$:

   $\forall_{nA}(\mathrm{set}_{xyz}(x -$ натуральное $\&\ y -$ натуральное $\&\ x < y\ \&\ y \leq n\ \&\ A(x,y,z)) = \mathrm{set}_{xyz}(x -$натуральное $\&\ y -$натуральное $\&\ x{+}1 \leq y\ \&\ x \leq n\ \&\ y \leq n\ \&\ A(x,y,z)))$

   Указатель "кортежпеременных(x25)" определяет идентификацию z с произвольным (возможно, пустым) списком переменных. A - функциональная переменная. Уровень срабатывания равен 1. Наконец, имеется прием для разрешения целочисленного линейного неравенства с константными коэффициентами:

   $\forall_{xmn}(x -$ натуральное $\rightarrow xn < m \leftrightarrow x \leq [m - 1/n])$

   Выражения m, n идентифицируются с натуральными константами. Неравенство входит в условие задачи на описание либо в посылку задачи на исследование. Заменяющий терм обрабатывается нормализаторами общей стандартизации, вычисляющими значение целой части. Уровень срабатывания приема равен 2.

3. Выдача ответа. Если в процессе решения задачи обнаруживается, что неизвестная x принимает натуральные значения, а единственное дополнительное условие на x - неравенство $x < a$, где a известно, то выдается ответ. Теорема приема имеет вид "$x < a\ \&\ x -$ натуральное". Заголовок приема - "ответзадачи". Предполагается, что выражение a неконстантное (иначе был бы выполнен переход к нестрогому неравенству для $a - 1$) и что имеется внешняя задача с несколькими неизвестными, из которой данная задача была получена временной фиксацией всех неизвестных, кроме x. Уровень срабатывания равен 4.

4. Разбор случаев по натуральной неизвестной, имеющей наибольшее значение. Прием возник при рассмотрении одной задачи на решение целочисленного неравенства специального вида:

   $\forall_{abcdmnkpqw}(a -$ натуральное $\&\ b -$ натуральное $\&\ c -$ натуральное $\&\ d -$ натуральное $\&\ 0 < m\ \&\ 0 < n\ \&\ 0 < k\ \&\ 0 < p\ \&\ 0 < q\ \&\ 0 < am + bn + ck + dp - abcdq\ \&\ w = m + n + k + p \rightarrow b \leq a\ \&\ c \leq a\ \&\ d \leq a\ \&\ bcdq < w\ \vee\ a \leq b\ \&\ c \leq b\ \&\ d \leq b\ \&\ acdq < w\ \vee\ a \leq c\ \&\ b \leq c\ \&\ d \leq c\ \&\ abdq < w\ \vee\ a \leq d\ \&\ b \leq d\ \&\ c \leq d\ \&\ abcq < w)$

Заголовок приема - "выводусловия". Десятый антецедент идентифицируется с условием задачи на описание. Выражения a, b, c, d содержат неизвестные, коэффициенты m, n, k, p, q суть целочисленные константы. Одиннадцатый антецедент выделен указателем "идентификатор", антецеденты с 5 по 9 - указателем "программа" и первые четыре антецедента - указателем "блокпроверок". Выводимая дизъюнкция снабжается комментарием "разборслучаев". Уровень срабатывания равен 7. Рекомендуется найти в разделе "элементарная алгебра" задачу, для которой был создан данный прием.

5. Разбор случаев для выражения, принимающего натуральные значения. Если содержащее неизвестные и принимающее натуральные значения выражение ограничено сверху не очень большой натуральной константой, то реализуется разбор случаев:

   $\forall_{xn}(x < n \ \& \ x - \text{натуральное} \rightarrow \exists_m(m \in \{1, \ldots, n-1\} \ \& \ x = m))$

   Прием имеет заголовок "выводусловия". Первый антецедент идентифицируется с условием задачи на описание. n - натуральная константа, меньшая 6. Выражение x содержит неизвестные. Указатель "или(фикс(0)фикс(0 2 1))" означает, что квантор существования будет развернут в дизъюнкцию $x = 1 \ \vee \ \ldots \ \vee \ x = n-1$. При этом первое подкванторное утверждение $m \in \{1, \ldots, n-1\}$ компилируется в процедуру, перечисляющую натуральные значения m. Для ускоренного отсечения ненужных контекстов введен фильтр, проверяющий, что задача имеет условие "натуральное(y)", где y - неизвестная, входящая в x. Второй антецедент обрабатывается проверочным оператором. Выведенная дизъюнкция сопровождается комментарием "разборслучаев". Уровень срабатывания приема равен 5.

Вывод следствий в посылках задачи на описание

При решении задачи на описание, ответ которой не должен зависеть от заданных переменных, может оказаться необходим вывод неравенств в посылках, дающих исключающие данную зависимость оценки. Здесь созданы следующие приемы:

1. Модуль суммы.

   $\forall_{abde}(0 < d - |a| \ \& \ 0 < e - |b| \rightarrow 0 < d + e - |a+b|)$

   Прием имеет заголовок "вывод". Оба антецедента идентифицируются с посылками задачи на описание. Существуют переменные, которые не должны входить в ответ. Выражение $|a+b|$ входит в посылку либо в условие. После применения приема веса всех термов задачи, содержащих это выражение, уменьшаются до 0. Уровень срабатывания равен 4.

2. Линейная комбинация неравенств.

   $\forall_{abcdef}(0 < a + bc \ \& \ 0 \leq d + ef \ \& \ b < 0 \ \& \ e < 0 \rightarrow bec + bef + ae + bd < 0)$

   $\forall_{abcdef}(0 < a + bc \ \& \ 0 \leq d + ef \ \& \ 0 < b \ \& \ 0 < e \rightarrow 0 < bec + bef + ae + bd)$

   Приемы инициируются при усмотрении в задаче выражения $c + f$, не расположенного под модулем. Оба слагаемых этого выражения содержат переменные, которые не должны войти в ответ. Два первых антецедента идентифицируются с посылками задачи. Они представляют собой неравенства для c и для f, взятых по отдельности. Коэффициенты a, b, d, e не содержат запрещенных переменных. Третий и четвертый антецеденты обрабатываются проверочными операторами. Ускоряющие фильтры сразу проверяют, что текущая сумма имеет не менее двух слагаемых с запрещенными переменными и что c, f неконстантные. Указатель "вариант" разрешает второму антецеденту идентифицироваться со строгим неравенством. После применения приема возникает неравенство для $c + f$, которое может оказаться полезным при исключении зависимости от запрещенных переменных. Уровень срабатывания равен 4.

Выдача ответа задачи на описание

Если относительно единственной неизвестной x в условиях задачи на описание остаются лишь два встречных строгих неравенства с известными противоположными частями, утверждение "x – число" и некоторое количество утверждений вида $x \neq a$, где a известно, то выдается ответ. При этом разрешается любой неполный непустой комплект перечисленных утверждений. Теорема приема имеет вид "явное(x1 набор(число(x1)меньше(x1 x2)меньше(x3 x1))набор(не(равно(x1 x4)))пустоеслово)". Чтобы усмотрение ответа могло произойти сразу, как только он появился, создается несколько экземпляров приема, каждый со своей точкой привязки. Заголовки этих приемов явно указывают точку привязки: "ответ(не(равно(x1 x4)))", "ответ(меньше(x3 x1))", "ответ(меньше(x1 x2))", "ответ(число(x1))". Уровни срабатывания равны 1.

Вся серия приемов создается автоматически после ввода теоремы приема. Нужно лишь нажать клавишу "а" (кир.), выбрать пункт "Справочники и сопровождающие их простейшие приемы" и нажать "курсор вправо". На экране начнут появляться перечисленные версии приемов, которые нужно будет сохранять нажатием "F3" и переходить к очередной версии нажатием "ш".

Обратный вывод в задаче с целью "независит"

$\forall_{abc}(0 < a + b \ \& \ 0 \leq c - b \to 0 < a + c)$

Прием имеет заголовок "подборзначений". Консеквент идентифицируется с условием задачи на описание. Выражение a имеет переменные, которые не должны входить в ответ, выражение c - не имеет. Первый антецедент идентифицируется с неравенством из контекста рассматриваемого условия, причем выражение b не содержит запрещенных переменных. Второй антецедент выделен указателем "подборзначений". Он определяет не содержащее запрещенных переменных неравенство, которое будет рассматриваться во вспомогательной задаче вместо $0 < a+c$. Уровень срабатывания равен 6.

Усмотрение избыточного условия отрицания равенства при наличии строгого неравенства

Если условие задачи на описание имеет вид отрицания равенства нулю некоторого выражения и не используется для сопровождения по о.д.з., то предпринимается попытка исключить его, выведя из имеющихся в контексте строгих неравенств. Начнем с двух простейших приемов такого типа:

$\forall_a(0 < a \to \neg(a = 0))$

$\forall_a(a < 0 \to \neg(a = 0))$

Приемы имеют заголовок "второйтерм". В обоих случаях отрицание равенства является условием задачи на описание, содержащим неизвестные и не используемым для сопровождения по о.д.з. Антецедент идентифицируется с неравенством из контекста. Уровень срабатывания равен 1. На приведенных теоремах созданы еще два приема, применяемых при редактировании ответа задачи на описание к отрицанию равенства нулю известного выражения a. Это отрицание уже может использоваться для сопровождения по о.д.з. Уровень срабатывания тоже равен 1.

Для несколько более общей ситуации созданы следующие два приема, которые, однако, имеют очень сильный ограничитель трудоемкости:

$$\forall_{abcd}(a < b \,\&\, \neg(c = 0) \,\&\, c(bc + d) \leq 0 \rightarrow \neg(ac + d = 0))$$

$$\forall_{abcd}(b < a \,\&\, \neg(c = 0) \,\&\, 0 \leq c(bc + d) \rightarrow \neg(ac + d = 0))$$

Здесь выражение a содержит неизвестные, выражения b, c, d - не содержат. Первый антецедент идентифицируется непосредственно, второй и третий обрабатываются проверочными операторами. Задача не должна иметь уравнений, пересекающихся по своим переменным с отрицанием равенства. Если отрицание равенства возникло при разборе случаев, причем имеется условное выражение, внутри которого встречается $ac + d$, то прием блокируется. Следующие два приема применяются в произвольных задачах:

$$\forall_{abc}(c < a - b \,\&\, 0 \leq c \rightarrow \neg(a = b))$$

$$\forall_{abc}(a - b < c \,\&\, c \leq 0 \rightarrow \neg(a = b))$$

Здесь c - константное выражение. Первый антецедент идентифицируется непосредственно, второй обрабатывается проверочным оператором. В специальных случаях отрицание равенства нулю известного подвыражения задачи на описание усматривается следующими приемами:

$$\forall_{abcde}(0 < -a + d \,\&\, 0 < a + d \,\&\, 0 < -b + e \,\&\, 0 < b + e \,\&\, |de| \leq |c| \rightarrow \neg(ab - c = 0))$$

$$\forall_{abcde}(0 < -a + d \,\&\, 0 < a + d \,\&\, 0 < -b + e \,\&\, 0 < b + e \,\&\, |de| \leq |c| \rightarrow \neg(-ab + c = 0))$$

Первые четыре антецедента, равносильные неравенствам $|a| < d, |b| < e$, имеются в контексте. Выражения c, d, e суть числовые константы. Пятый антецедент проверяется непосредственным вычислением. Условие не должно использоваться для сопровождения по о.д.з. Уровень срабатывания равен 2. При редактировании ответа задачи на описание используется еще один прием:

$$\forall_{abc}(0 < b + a \,\&\, a + c < 0 \rightarrow \neg(b = c))$$

Отрицание равенства не содержит неизвестных и является условием задачи. b - переменная; a, c - константы. Первый антецедент идентифицируется непосредственно, второй обрабатывается проверочным оператором. Уровень срабатывания равен 3. В заключение приведем прием, применяемый в задачах любого типа, кроме задач на исследование:

$$\forall_{abc}(0 \leq c - b \,\&\, c < a \rightarrow \neg(a = b))$$

Второй антецедент идентифицируется непосредственно, первый обрабатывается проверочным оператором. Выражения b, c либо оба константные, либо равны. Допускается применение приема к утверждениям, используемым для сопровождения по о.д.з. Уровень срабатывания равен 0.

Попытка опровергнуть уравнение путем доказательства строгого неравенства

Если уравнение решить не удается, то на достаточно высоком уровне предпринимается попытка усмотреть его ложность, доказав строгое неравенство для разности частей:

$\forall_{ab}(0 < a - b \to \neg(a = b))$

Прием имеет заголовок "второйтерм". Равенство $a = b$ представляет собой условие задачи на описание, содержащее неизвестные. Усматривается, что хотя бы одна из частей равенства имеет числовое значение. Задача не должна иметь цели "исследовать", означающей, что требуется предпринять общее исследование свойств неизвестных. Отбрасываются случаи, когда в одной части равенства находится неизвестная, а другая часть известна. При решении дифференциальных уравнений прием не применяется. Предполагается также, что не имеет места этап редактирования ответа. Прием снабжен слабым ограничителем трудоемкости. Антцедент обрабатывается задачей на доказательство, решаемой с максимальным уровнем 5. Уровень срабатывания приема равен 8. Создана еще одна версия приема, ориентированная на редко встречающийся случай задач с целью "не". Они возникают при попытке явного разрешения равенства, находящегося под отрицанием в некоторой внешней задаче. В этой версии уровень решения задачи на доказательство повышен до максимального уровня текущей задачи. Наконец, по той же теореме создан прием для усмотрения ложности сопровождающего условия на известные параметры, имеющего вид равенства. Перед попыткой применения проверяется наличие в посылках неравенства, пересекающегося по своим параметрам с рассматриваемым равенством. Уровень срабатывания здесь равен 5.

Усмотрение положительности параметра в задаче на исследование, имеющей цель "известно"

Напомним, что задачи на исследование с целью "известно" аналогичны геометрическим задачам на вычисление. Иногда в таких задачах бывает полезно своевременно усмотреть и зафиксировать положительность численного параметра, чтобы воспользоваться ею на более поздних этапах решения, когда она станет уже неочевидной. Для этого служит следующий прием вывода:

$\forall_{ab}(a = b \;\&\; 0 < b \to 0 < a)$

Переменная a идентифицируется с известным численным параметром задачи. Первый антецедент представляет собой посылку, явно определяющую a через другие переменные задачи. Обычно последние не известны; например, a может быть известной длиной отрезка с неизвестными концами. Второй антецедент обрабатывается проверочным оператором. Перед попыткой применения приема проверяется, что неизвестные внешней задачи на описание еще не найдены и что пока отсутствует посылка $0 < a$. Введен ограничитель трудоемкости. Уровень срабатывания равен 0.

Объединение случаев с противоположными неравенствами

Если в задаче на исследование возникает дизъюнкция, определяющая для каждого подслучая значение неизвестной, то она преобразуется в равенство неизвестной условному выражению:

$\forall_{abcdx}(x = a \ \& \ 0 < c - d \ \vee \ x = b \ \& \ 0 \leq d - c \leftrightarrow x = (a \text{ при } 0 < c - d, \text{ иначе } b))$

Дизъюнкция не обязана быть корневой; задача должна иметь цель "известно". Переменная x идентифицируется с неизвестной внешней задачи на описание, выражения a, b, c, d известны. Уровень срабатывания равен 1.

Еще два приема, исключающих дизъюнкцию, применяются в задачах на описание, имеющих цель "и":

$\forall_{ab}(0 < a \ \& \ b < 0 \ \vee \ a < 0 \ \& \ 0 < b \leftrightarrow ab < 0)$

$\forall_{ab}(0 < a \ \& \ 0 < b \ \vee \ a < 0 \ \& \ b < 0 \leftrightarrow 0 < ab)$

Дизъюнкция не содержит неизвестных и является условием задачи. Имеет место этап редактирования ответа. Напомним, что цель "и" означает получение ответа в виде конъюнкции элементарных утверждений. Применение приема позволяет преобразовать дизъюнкцию в такое утверждение. Уровень срабатывания равен 2.

Усмотрение отрицания равенства

$\forall_{ab}(a < b \rightarrow \neg(a = b))$

Этот совсем простой прием имеет заголовок "второйтерм"; его антецедент идентифицируется с утверждением из контекста. Уровень срабатывания равен 1.

Проверочный оператор "усмменьше"

Проверочный оператор "усмменьше" используется для быстрого усмотрения истинности строгих неравенств. Число уровней срабатывания его приемов равно 4. Некоторые из перечисляемых ниже приемов относятся к смежным разделам (геометрия, физика и т.п.).

1. Непосредственное усмотрение. Как и любой проверочный оператор, прежде всего оператор "усмменьше" обращается к процедуре "стандследствие". Она ищет в посылках проверяемое утверждение либо применяет ряд простейших средств. В данном случае эти средства сводятся к получению численных оценок для установления знака константного выражения. Используется оператор "числоценка", учитывающий 4 знака после запятой и дающий гарантированную верхнюю либо нижнюю оценку.

2. Переход к неравенству с нулевой частью. Если проверяемое неравенство не имеет нуля в одной из своих частей, то оно сразу же преобразуется к виду неравенства с нулем в левой части:

   $\forall_{ab}(0 < a - b \rightarrow b < a)$

   Выражения a, b ненулевые. Антецедент обрабатывается проверочным оператором. Таким образом, реализуется рекурсивное обращение, в котором новым входным данным служит $0 < a - b$. Указатель "спуск" блокирует попытки применения других приемов, если это рекурсивное обращение окзалось неудачным. Прием применяется на уровне 1.

3. Устранение минуса.

   $\forall_{a}(a < 0 \rightarrow 0 < -a)$

   $\forall_{a}(0 < a \rightarrow -a < 0)$

   Фильтры и указатели аналогичны предыдущему приему.

4. Транзитивность с использованием неравенства из посылок, имеющего нулевую часть. Так как неравенства в посылках приводятся к виду с нулевой левой либо правой частью, то обычная транзитивность несколько модифицируется. Далее будут приведены также несколько более общие приемы, использующие линейные комбинации неравенств. Однако, они будут сопровождаться существенными ограничениями: обращения к оператору "усмменьше" выполняются очень часто, и даже небольшие задержки в его работе нежелательны.

   $\forall_{ab}(b - a < 0 \ \& \ 0 \leq b \rightarrow 0 < a)$

   $\forall_{ab}(0 < a + b \ \& \ b \leq 0 \rightarrow 0 < a)$

   $\forall_{ab}(a + b < 0 \ \& \ 0 \leq b \rightarrow a < 0)$

   $\forall_{ab}(0 \leq b - a \ \& \ b < 0 \rightarrow a < 0)$

   $\forall_{ab}(0 < b - a \ \& \ b \leq 0 \rightarrow a < 0)$

   $\forall_{ab}(0 \leq a + b \ \& \ b < 0 \rightarrow 0 < a)$

   $\forall_{ab}(a + b \leq 0 \ \& \ 0 < b \rightarrow a < 0)$

   Выражение a неконстантное. Первый антецедент идентифицируется с посылкой, второй обрабатывается проверочным оператором. Прием блокируется при наличии комментария "перестановка", указывающего, что была предпринята попытка перенести нуль в противоположную часть неравенства для разности. Прием, предпринимающий такую попытку, описывается ниже. Введен средний ограничитель трудоемкости. Уровень срабатывания равен 2. Приведем еще несколько приемов аналогичного типа, у которых ненулевая часть имеет три слагаемых:

   $\forall_{abc}(0 < a - b \ \& \ 0 < b - c \rightarrow 0 < 2a - b - c)$

   $\forall_{abc}(0 < a - b \ \& \ 0 < b - c \rightarrow 2c - a - b < 0)$

   $\forall_{abc}(0 \leq a - b \ \& \ c < 0 \rightarrow b - a + c < 0)$

   В первых двух случаях антецеденты идентифицируются с посылками, в последнем случае второй антецедент обрабатывается проверочным оператором. Уровни срабатывания равны 3.

5. Усмотрение знака суммы по транзитивности. В разделе собраны приемы, аналогичные приемам предыдущего раздела, но относящиеся только к неравенствам, у которых в одной части находится сумма, а в другой - ноль.

   $\forall_{abc}(b - c < 0 \ \& \ 0 \leq a + b \rightarrow 0 < a + c)$

   $\forall_{abc}(0 < a + b \ \& \ 0 \leq c - b \rightarrow 0 < a + c)$

   $\forall_{abc}(a + c < 0 \ \& \ 0 \leq c - b \rightarrow a + b < 0)$

   $\forall_{abc}(0 < c - b \ \& \ a + c \leq 0 \rightarrow a + b < 0)$

   $\forall_{abc}(0 < a + c \ \& \ b + c \leq 0 \rightarrow 0 < a - b)$

   $\forall_{ab}(0 < a \ \& \ 0 \leq b \rightarrow 0 < a + b)$

   Во всех случаях один антецедент идентифицируется непосредственно, другой обрабатывается проверочным оператором. На первой, третьей и четвертой теоремах созданы по два приема, отличающиеся выбором непосредственно идентифицируемого антецедента. У первой и шестой теорем непосредственно идентифицируется первый антецедент, у пятой - второй. Общее слагаемое у текущего неравенства и неравенства из посылок определяется как пересечение групп операндов. За исключением последнего приема, оно должно быть неконстантным, причем либо оставшаяся часть слагаемых текущего неравенства должна быть константной, либо неравенство, обрабатываемое проверочным оператором, должно иметь или всего два слагаемых, или не более одного неконстантного слагаемого. Эти эвристические ограничения появились в результате попыток ускорения оператора. Кроме того, введен умеренный общий ограничитель трудоемкости приемов. Уровни срабатывания равны 3.

6. Транзитивность с использованием неравенства из посылок, не имеющего нулевой части. Если проверка выполняется относительно контекста условия задачи на описание, то среди посылок проверочного оператора могут встретиться неравенства с неизвестными, обе части которых ненулевые. Неравенства с ненулевыми частями в определенных случаях могут встретиться и среди посылок задачи. Для таких неравенств созданы свои приемы усмотрения по транзитивности:

   $\forall_{abc}(a < b \;\&\; b + c \leq 0 \to a + c < 0)$

   $\forall_{abc}(c < b \;\&\; 0 \leq c - a \to a - b < 0)$

   $\forall_{abcd}(a \leq b + d \;\&\; 0 < c - d \to 0 < b + c - a)$

   $\forall_{abc}(c \leq a \;\&\; 0 < c + b \to 0 < a + b)$

   $\forall_{abc}(a < b \;\&\; 0 \leq c - b \to 0 < c - a)$

   $\forall_{abc}(a \leq b \;\&\; b + c < 0 \to a + c < 0)$

   $\forall_{abc}(c \leq b \;\&\; 0 < c - a \to a - b < 0)$

   $\forall_{abc}(c < a \;\&\; 0 \leq c + b \to 0 < a + b)$

   $\forall_{abc}(a \leq b \;\&\; 0 < c - b \to 0 < c - a)$

   Во всех этих случаях неравенство с ненулевыми частями берется в посылках, второе неравенство обрабатывается проверочным оператором. Слагаемые проверяемого неравенства, общие с неравенством из посылок, неконстантные; противоположная часть неравенства из посылок ненулевая. Третий прием имеет уровень срабатывания 1, остальные - уровень 2. Добавлены еще два приема:

   $\forall_{ab}(a < -b \;\&\; 0 \leq a \to b < 0)$

   $\forall_{mnk}(n + k \leq m \;\&\; 0 < k \to 0 < m - n)$

   Первый из них срабатывает на уровне 2 без каких-либо ограничений. Во втором m, n должны быть переменными, не входящими в выражение k. Его уровень срабатывания равен 3. Как и везде в решателе, данный комплект приемов сложился "стихийно", по мере проработки задач. Он представляет собой лишь исходный материал для развития техники оптимизации решателей. Заметим, что пополнение решателя приемами "из общих соображений", не подкрепленное анализом задач, легко приводит к появлению избыточных приемов, без которых в действительности следует обходиться. Либо решение будет обычно проходить по руслу, исключающему возможность срабатывания приема, либо прием начнет нарушать сложившиеся стандарты и замедлять или вовсе блокировать решение.

7. Попытка одновременного изменения знака у всех слагаемых. При доказательстве неравенства для разности предпринимается попытка доказать встречное неравенство для противоположной разности:

$\forall_{ab}(b-a<0\to 0<a-b)$

$\forall_{ab}(0<a-b\to b-a<0)$

Антецедент выполняет рекурсивное обращение к оператору "усмменьше". Оно сопровождается комментарием "перестановка", блокирующим встречную попытку изменения знака, а также ряд приемов, которые могли бы сработать и без данной попытки. Если попытка не увенчалась успехом, оператор продолжает обрабатывать неравенство другими приемами. Перед обращением проверяется, что хотя бы одно из выражений a, b неконстантное. Уровень срабатывания равен 2. Созданы еще два аналогичных приема, которые не выполняют рекурсивное обращение, а используют готовую посылку:

$\forall_{abc}(a(b-c)<0\to 0<a(c-b))$

$\forall_{abc}(0<a(b-c)\to a(c-b)<0)$

Уровень срабатывания этих приемов равен 2.

8. Установление знака суммы путем установления знаков слагаемых.

   $\forall_{ab}(a<0\ \&\ b\leq 0\to a+b<0)$

   $\forall_{ab}(0<a\ \&\ 0\leq b\to 0<a+b)$

   Предполагается, что сумма неконстантная, так как знак константных сумм находится непосредственной численной оценкой. Исключение делается для констант, содержащих единицы измерения. Оба антецедента обрабатываются проверочными операторами. Указатель "операнд(x1 ...)" определяет идентификацию переменной a с некоторым слагаемым суммы. Таким образом просматриваются все слагаемые, пока не удается найти слагаемое заданного знака (отрицательное в первом приеме и положительное во втором). После этого реализуется попытка доказать нестрогое неравенство для суммы остальных слагаемых. Чтобы в случае неудачи не происходило продолжение перечисления слагаемых a, введен указатель "откат(1)". При наличии комментария "перестановка" прием блокируется. Обращения к проверочным операторам, обрабатывающим антецеденты, сопровождаются комментарием "множители", блокирующим попытки разложения на множители выражений a, b (см. ниже соответствующий прием). Первый прием имеет уровень срабатывания 3, второй - уровень 2.

9. Усмотрение знака произведения.

   Прежде всего, предпринимается попытка усмотрения положительности всех сомножителей:

   $\forall_{ab}(0<b\ \&\ 0<a\to 0<ab)$

   Указатель "дистрибразвертка(фикс(0 2))" означает, что компилятор учитывает только один из антецедентов (например, первый), используя его как "шаблон" для обработки каждого сомножителя. Так как антецеденты выделены указателем "блокпроверок", то для проверки положительности сомножителей будут выполняться рекурсивные обращения к оператору "усмменьше". Прием срабатывает на уровне 1. Следующие приемы выделяют один сомножитель определенного знака, а затем анализируют оставшееся произведение:

   $\forall_{ab}(0<b\ \&\ a<0\to ab<0)$

   $\forall_{ab}(b<0\ \&\ a<0\to 0<ab)$

$\forall_{ab}(0 < b \mathbin{\&} 0 < a \rightarrow 0 < ab)$

Оба антецедента обрабатываются проверочным оператором. Указатель "спуск(1)" предотвращает попытки альтернативных действий, если удалось реализовать первый антецедент. При неудачной попытке проверки второго антецедента сразу будет выдан "отказ". Уровень срабатывания равен 2.

10. Устранение дроби. Проверка положительности или отрицательности дроби сводится к аналогичной проверке для произведения:

    $\forall_{ab}(0 < ab \rightarrow 0 < a/b)$

    $\forall_{ab}(ab < 0 \rightarrow a/b < 0)$

    Антецедент обрабатывается проверочным оператором. Для первого приема созданы две версии. Если посылки не имеют утверждения о стремлении некоторого параметра к заданной величине, то применяется версия, снабженная указателем "спуск". В ней блокируются альтернативные попытки усмотрения знака дроби. Иначе - блокировки альтернативных попыток не происходит. Второй прием имеет одну версию, с указателем "спуск". Для числовой дроби приемы блокируются. Уровень срабатывания приемов с указателем "спуск" равен 1, приема без этого указателя - 3.

11. Устранение степени.

    $\forall_{ab}(0 < a \rightarrow 0 < a^b)$

    Антецедент обрабатывается проверочным оператором. Уровень срабатывания равен 1. Для рационального показателя степени введены еще два приема:

    $\forall_{ab}(b - \mathrm{rational} \mathbin{\&} \neg(\text{знаменатель}(b) - \mathrm{even}) \mathbin{\&} \neg(\text{числитель}(b) - \mathrm{even}) \mathbin{\&} a < 0 \rightarrow a^b < 0)$

    $\forall_{ab}(b - \mathrm{rational} \mathbin{\&} \text{числитель}(b) - \mathrm{even} \mathbin{\&} \neg(a = 0) \rightarrow 0 < a^b)$

    Антецеденты обрабатываются проверочными операторами. Первый прием имеет указатель "спуск", блокирующий альтернативные попытки. Второй прием имеет указатель "спуск(2)", блокирующий альтернативные попытки после усмотрения четности числителя показателя степени. Уровни срабатывания равны 1.

12. Разность степеней с одинаковыми показателями. Степени разделяются в разных частях неравенства и используется монотонность степенной функции:

    $\forall_{abc}(0 < c \mathbin{\&} 0 \leq a \mathbin{\&} 0 \leq b \mathbin{\&} a - b < 0 \rightarrow a^c - b^c < 0)$

    $\forall_{abc}(0 < c \mathbin{\&} 0 \leq a \mathbin{\&} 0 \leq b \mathbin{\&} 0 < a - b \rightarrow 0 < a^c - b^c)$

    Показатель степени - константа. Антецеденты обрабатываются проверочными операторами. Если удается установить истинность первых трех из них, то попытки использовать альтернативные приемы блокируются, и все сводится к обработке четвертого антецедента. Это обеспечивается указателем "спуск(1 2 3)". Если показатель степени - простая дробь с нечетным знаменателем, то прием блокируется, так как в этом случае используются другие приемы:

    $\forall_{abc}(c - \mathrm{rational} \mathbin{\&} 0 < c \mathbin{\&} \neg(\text{числитель}(c) - \mathrm{even}) \mathbin{\&} a - b < 0 \rightarrow a^c - b^c < 0)$

    $\forall_{abc}(c - \mathrm{rational} \mathbin{\&} 0 < c \mathbin{\&} \neg(\text{числитель}(c) - \mathrm{even}) \mathbin{\&} 0 < a - b \rightarrow 0 < a^c - b^c)$

Все четыре приема имеют уровень срабатывания 2. Создан еще один прием, использующий равенство для разности степеней, имеющееся в посылках:

$\forall_{abcd}(a^d - b^d = c \;\&\; 0 < c \;\&\; 0 \le a \;\&\; 0 \le b \to 0 < a - b)$

Неравенства в антецедентах обрабатываются проверочными операторами. Уровень срабатывания равен 3.

13. Неравенства с логарифмами. Усматривается знак логарифма:

    $\forall_{ab}(0 < a - 1 \;\&\; 0 < b - 1 \to 0 < \log_a b)$

    $\forall_{ab}(0 < a - 1 \;\&\; 0 < 1 - b \to \log_a b < 0)$

    Антецеденты обрабатываются проверочными операторами. После установления истинности первого из них попытки применения других приемов блокируются. Уровень срабатывания равен 2. Для двойного натурального логарифма добавлен отдельный прием:

    $\forall_a(0 < a - e \to 0 < \ln \ln a)$

    Здесь уровень срабатывания равен 1.

14. Усмотрение положительности модуля.

    $\forall_a(a - \text{число} \;\&\; \neg(a = 0) \to 0 < |a|)$

    Попытки применения альтернативных приемов блокируются, уровень срабатывания равен 1. В этом же разделе представлен прием, относящийся к модулю комплексных чисел:

    $\forall_a(\neg(a = 0) \to 0 < |a|)$

    Заметим, что мы вынуждены рассматривать здесь приемы из других разделов (в данном случае - из комплексного анализа), так как они попали в пакет "усм-меньше" и нужны для быстрого усмотрения истинности числовых неравенств. Заметим также, что модули в двух приемах разные, хотя и прорисовываются формульным редактором одинаково. При просмотре теорем в скобочной записи становится видно, что первая теорема использует вещественнозначный символ "модуль", а вторая - символ "Модуль" для комплексного случая.

15. Неравенства с радикалами. Усматривается положительность или отрицательность некоторых часто встречающихся сумм с радикалом:

    $\forall_{ab}(0 < b \to 0 < \sqrt{a^2 + b} + a)$

    $\forall_{ab}(0 < b \to 0 < \sqrt{a^2 + b} - a)$

    $\forall_{abc}(2ab + b^2 - c < 0 \to a + b - \sqrt{a^2 + c} < 0)$

    $\forall_{abc}(0 < c \to 0 < a - b + \sqrt{a^2 + b^2 - 2ab + c})$

    $\forall_{abc}(\neg(b = 0) \;\&\; 0 \le c + 1 \;\&\; 0 < a \to 0 < a + \sqrt{(a + b)(a - b)c})$

    Первые три приема срабатывают на уровне 1, последние два - на уровне 3. Антецеденты обрабатываются проверочными операторами.

16. Квадратный трехчлен с числовыми коэффициентами, имеющий отрицательный дискриминант.

    $\forall_{abcd}(0 < a \;\&\; b^2 - 4ac < 0 \to 0 < ad^2 + bd + c)$

    $\forall_{abcd}(a < 0 \;\&\; b^2 - 4ac < 0 \to ad^2 + bd + c < 0)$

Переменные a, b, c идентифицируются с десятичными константами. Оба антецедента выделены указателем "программа". Дополнительно созданы приемы для однородного трехчлена второй степени:

$\forall_{pqrbc}(p < 0 \ \& \ r^2 - 4pq < 0 \ \& \ \neg(c = 0) \to pb^2 + qc^2 + rbc < 0)$

$\forall_{pqrbc}(0 < p \ \& \ r^2 - 4pq < 0 \ \& \ \neg(c = 0) \to 0 < pb^2 + qc^2 + rbc)$

Здесь коэффициенты p, q, r идентифицируются с десятичными константами, b, c - с неконстантными выражениями. Первые два антецедента выделены указателем "программа", третий - обрабатывается проверочным оператором. Все четыре приема имеют уровень срабатывания 2.

17. Линейная комбинация косинуса и косинуса двойного аргумента.

$\forall_{abcd}(c = 2d \ \& \ 0 < a + b \ \& \ b < 0 \ \& \ 0 \le d \ \& \ 0 \le \pi/2 - d \to 0 < a\cos d + b\cos c)$

Коэффициенты a, b идентифицируются с десятичными константами. В посылках имеется неравенство, содержащее число π. Первый антецедент, выделенный указателем "идентификатор", сравнивает аргументы c, d. При этом произведение $2d$ обрабатывается нормализатором общей стандартизации "нормумножение" и нормализатором раскрывания скобок "стандплюс". Второй и третий антецеденты выделены указателем "программа". Четвертый и пятый - обрабатываются проверочными операторами. Для усиления проверки пятого антецедента создана дополнительная версия приема, в которой предпринимается попытка разложить правую часть на множители. Эта попытка сопровождается сильным ограничителем трудоемкости. Обе версии срабатывают на уровне 4.

18. Усмотрение знака суммы при помощи линейных комбинаций с посылками. Если в посылках имеется неравенство, ненулевая часть которого отличается от ненулевой части доказываемого неравенства лишь умножением на константный множитель и добавлением константного слагаемого, то делается попытка вывести одно неравенство из другого:

$\forall_{abcde}(b + ce < 0 \ \& \ 0 \le e(ae - bd) \ \& \ ed < 0 \to 0 < a + cd)$

$\forall_{abcde}(0 \le b + ce \ \& \ 0 < e(ae - bd) \ \& \ 0 < ed \to 0 < a + cd)$

$\forall_{abcde}(0 < b + ce \ \& \ e(ae - bd) \le 0 \ \& \ ed < 0 \to a + cd < 0)$

$\forall_{abcde}(b + ce \le 0 \ \& \ e(ae - bd) < 0 \ \& \ 0 < ed \to a + cd < 0)$

$\forall_{abcde}(b + ce \le 0 \ \& \ 0 < e(ae - bd) \ \& \ ed < 0 \to 0 < a + cd)$

$\forall_{abcde}(0 < b + ce \ \& \ 0 \le e(ae - bd) \ \& \ 0 < ed \to 0 < a + cd)$

$\forall_{abcde}(0 \le b + ce \ \& \ e(ae - bd) < 0 \ \& \ ed < 0 \to a + cd < 0)$

$\forall_{abcde}(b + ce < 0 \ \& \ e(ae - bd) \le 0 \ \& \ 0 < ed \to a + cd < 0)$

В этих приемах переменные a, b, d, e идентифицируются с константными выражениями, переменная c - с неконстантным выражением. Первый антецедент берется из списка посылок, следующие два - обрабатываются проверочными операторами. Либо a, либо b должно быть отлично от 0. Указатель "набороперандов(...)" позволяет идентифицировать c не при помощи сравнительно трудоемкого оператора "алгебрпересечение", а просто как пересечение групп сомножителей одночленов ce, cd. При наличии комментария "перестановка" прием блокируется. Уровень срабатывания равен 2. Если посылки содержат неравенство для переменной x, предпринимается попытка использовать его для усмотрения знака выражения $ax + b$, где a, b - константы:

$\forall_{abcx}(0 < a \ \& \ c < x \ \& \ 0 \leq ac + b \rightarrow 0 < ax + b)$

$\forall_{abcx}(a < 0 \ \& \ x < c \ \& \ 0 \leq ac + b \rightarrow 0 < ax + b)$

$\forall_{abcx}(0 < a \ \& \ x < c \ \& \ ac + b \leq 0 \rightarrow ax + b < 0)$

$\forall_{abcx}(a < 0 \ \& \ c < x \ \& \ ac + b \leq 0 \rightarrow ax + b < 0)$

Второй антецедент берется в посылках, остальные обрабатываются проверочными операторами. Уровень срабатывания равен 4.

19. Попытка разложения на множители. Разложение на множители является достаточно трудоемкой процедурой и должно выполняться до обращения к проверочному оператору. Однако, в простейшем случае усмотрения положительности суммы двух слагаемых попытка разложения на множители все же применяется:

    $\forall_{ab}(b = a \ \& \ 0 < b \rightarrow 0 < a)$

    Выражение a должно иметь вид суммы с двумя слагаемыми. Ни одно из этих слагаемых не должно иметь вида дроби. Первый антецедент, выделенный указателем "идентификатор", обращается к оператору "видумножение". Результат b не должен иметь вид суммы. Второй антецедент обрабатывается проверочным оператором. Ряд комментариев, включая комментарий "перестановка", блокируют попытку применения приема. Уровень срабатывания равен 3.

20. Использование отрицания равенства нулю, содержащегося в посылках. В данном случае достаточно проверить нестрогое неравенство:

    $\forall_{a}(\neg(a = 0) \ \& \ 0 \leq a \rightarrow 0 < a)$

    $\forall_{a}(\neg(a = 0) \ \& \ a \leq 0 \rightarrow a < 0)$

    $\forall_{a}(\neg(a = 0) \ \& \ \neg(a < 0) \rightarrow 0 < a)$

    $\forall_{a}(\neg(a = 0) \ \& \ \neg(0 < a) \rightarrow a < 0)$

    В первых двух случаях второй антецедент обрабатывается проверочным оператором, в последних двух случаях - берется в посылках. Заметим, что общая стандартизация переводит отрицание строгого неравенства в нестрогое обратное неравенство, однако возможны случаи, когда посылки будут не стандартизированы. Первые два приема имеют указатель "спуск", блокирующий альтернативные попытки сразу после идентификации отрицания равенствва. Уровень срабатывания последних приемов равен 1, первых - 2.

21. Использование нестрогого неравенства, содержащегося в посылках. В этом случае проверка сводится к усмотрению отрицания равенства нулю:

    $\forall_{a}(\neg(a = 0) \ \& \ 0 \leq a \rightarrow 0 < a)$

    $\forall_{a}(\neg(a = 0) \ \& \ a \leq 0 \rightarrow a < 0)$

    Первый антецедент обрабатывается проверочным оператором. Обращение сопровождается комментариями (легковидеть усмменьше 0 a), (легковидеть усмменьше a 0), блокирующими попытки усмотрения отличия a от нуля путем установления положительности либо отрицательности. Уровень срабатывания равен 1.

22. Выделение суммы константных слагаемых. Если несколько константных слагаемых имеют различные знаки, то целесообразно сгруппировать их и постараться усмотреть знак всей группы:

    $\forall_{ab}(a < 0 \;\&\; b \leq 0 \to a + b < 0)$

    $\forall_{ab}(0 < a \;\&\; 0 \leq b \to 0 < a + b)$

    Указатель "перечень(x1 константа(x1))" задает указанную группировку константных слагаемых в выражение a. Проверяется, что число слагаемых более одного и что остаток суммы b ненулевой. Оба антецедента обрабатываются проверочными операторами. Уровень срабатывания равен 3.

23. Использование неравенства специального вида, содержащегося в посылках.

    (a) Простейшие случаи.

    $\forall_{ab}(a \leq b \;\&\; b < 0 \to a < 0)$

    Первый антецедент содержится в посылках, причем правая часть его является константным выражением. Допускается версия строгого неравенства. Второй антецедент обрабатывается проверочным оператором. Уровень срабатывания равен 3.

    $\forall_{ab}(0 < a \;\&\; b = a \to 0 < b)$

    В посылках содержится второй антецедент, причем выражение b неконстантное. Первый антецедент обрабатывается проверочным оператором. Уровень срабатывания равен 2.

    $\forall_{ab}(a < b \to 0 < b - a)$

    Антецедент содержится в посылках. Указатель "повтор(1)" разрешает использовать эту посылку, даже если она содержится в комментарии (исключение ...), перечисляющем посылки, использованные надоператорами. Уровень срабатывания равен 3.

    $\forall_{ab}(-a < b \to 0 < a + b)$

    $\forall_{ab}(a < -b \to a + b < 0)$

    Уровень срабатывания равен 3.

    $\forall_{abc}(a < c \;\&\; 0 < b - c \to 0 < b - a)$

    Первый антецедент берется из посылок, причем допускается версия нестрогого неравенства. Выражения b, c константные, c отлично от нуля. Второй антецедент обрабатывается проверочным оператором. Уровень срабатывания равен 4.

    $\forall_{abcx}(x < c \;\&\; 0 < a \;\&\; ac + b \leq 0 \to ax + b < 0)$

    $\forall_{abcx}(c < x \;\&\; a < 0 \;\&\; ac + b \leq 0 \to ax + b < 0)$

    Выражение x представляет собой переменную, выражения a, b, c константные. Первый антецедент берется в посылках, прочие - обрабатываются проверочными операторами. Выражение c отлично от нуля. Уровень срабатывания равен 3.

    (b) Усмотрение неравенства с модулем.

    $\forall_{ab}(0 < a + b \;\&\; 0 < a - b \to 0 < a - |b|)$

    $\forall_{ab}(a + b < 0 \to b - |a| < 0)$

    Антецеденты берутся из посылок. Уровень срабатывания равен 3.

(c) Посылка с арккосинусом. Используется верхняя оценка π для модуля арккосинуса:

$\forall_{abcdpq}(b < 0 \;\&\; 0 < pq \;\&\; 0 \leq a + b\pi p/q \;\&\; c < p \arccos d/q \to 0 < a + bc)$

Выражение c неконстантное. Первые три антецедента обрабатываются проверочными операторами, четвертый - берется в посылках. Уровень срабатывания равен 3.

(d) Неравенство для разности дроби и единицы.

$\forall_{ab}(0 < b \;\&\; a/b - 1 < 0 \to a - b < 0)$

Первый антецедент обрабатывается проверочным оператором. Уровень срабатывания равен 3.

(e) Два неравенства для произведения.

$\forall_{abc}(0 < ab \;\&\; 0 < ac \to 0 < bc)$

Антецеденты берутся в посылках. Уровень срабатывания равен 3.

(f) Использование неравенства для произведения.

$\forall_{ab}(0 < ab \;\&\; 0 < a \to 0 < b)$

$\forall_{ab}(0 < ab \;\&\; a < 0 \to b < 0)$

Антецеденты берутся в посылках. Указатель "набороперандов(фикс(1 2))" позволяет вместо оператора "алгебрпересечение" использовать проверку включения групп сомножителей. Уровень срабатывания равен 3. Заметим, что приемы относятся к случаю ненормализованных посылок (иначе первое неравенство было бы заменено на неравенство для b).

(g) Домножение на знаменатель слагаемого из неравенства в посылках.

$\forall_{abcd}(0 < c \;\&\; 0 < a + b/c \;\&\; 0 \leq d - b \to 0 < ac + d)$

Второй антецедент берется из посылок, первый и третий обрабатываются проверочными операторами. Знаменатель c должен представлять собой десятичную константу, выражение a неконстантное. Для третьего антецедента введен сильный ограничитель трудоемкости. Уровень срабатывания равен 3.

(h) Использование кванторной посылки.

$\forall_{afg}(f(a) \;\&\; \forall_x(f(x) \to 0 < g(x)) \to 0 < g(a))$

Второй антецедент идентифицируется с кванторной импликацией из посылок. При этом $f(x)$ - функциональная переменная, идентифицируемая с конъюнкцией антецедентов, а $g(x)$ - не функциональная переменная, т.е. выражение вида "значение(g x)". Первый антецедент выделен указателем "очевидно". Он обрабатывается проверочными операторами, заголовки которых определяются в зависимости от вида конкретных идентифицированных с $f(x)$ утверждений. Уровень срабатывания равен 3.

(i) Использование принадлежности множеству, заданному описателем "класс".

$\forall_{axP}(x \in a \;\&\; a \subseteq \mathrm{set}_y(0 < y \;\&\; P(y)) \to 0 < x)$

Оба антецедента берутся в посылках. $P(x)$ - функциональная переменная. Уровень срабатывания равен 3.

24. Геометрические величины. Для усмотрения строгих неравенств, связанных с геометрическими величинами, в пакетном операторе предусмотрены следующие приемы:

(a) Положительность расстояния.

$\forall_{AB}$(разныеточки$(A, B) \rightarrow 0 < l(AB))$

Здесь и далее $l(AB)$ обозначает расстояние между точками A, B. Чтобы усматривать различие точек, в геометрических разделах решателя создан проверочный оператор "разныеточки". Для удобства компиляции обращений к нему, в теоремах приемов вместо терма "$\neg(A = B)$" берется терм "разныеточки(A, B)". Так как, в свою очередь, оператор "разныеточки" может обратиться к оператору "усмменьше", для предотвращения зацикливания используется комментарий "разныеточки". Он вводится оператором "разныеточки". Уровень срабатывания равен 2.

$\forall_{ABa}(l(AB) = a \ \& \ \neg(A = B) \rightarrow 0 < a)$

В этом случае оба антецедента берутся непосредственно из посылок. Уровень срабатывания равен 1.

(b) Положительность площади.

$\forall_{ABCD}$(ромб$(ABCD) \rightarrow 0 < S($фигура$(ABCD)))$

$\forall_{ABCD}$(параллелограмм$(ABCD) \rightarrow 0 < S($фигура$(ABCD)))$

$\forall_{ABCD}(\triangle(ABC) \rightarrow 0 < S($фигура$(ABC)))$

$\forall_{ABCD}$(трапеция$(ABCD) \rightarrow 0 < S($фигура$(ABCD)))$

$\forall_{ABCD}$(квадрат$(ABCD) \rightarrow 0 < S($фигура$(ABCD)))$

$\forall_{ABCD}$(прямоугольник$(ABCD) \rightarrow 0 < S($фигура$(ABCD)))$

Уровень срабатывания приемов равен 2.

$\forall_{ABC}$(разныепрямые(прямая(AB), прямая(AC)) &
разныеточки(A, B) & разныеточки$(A, C) \rightarrow 0 < S($фигура$(ABC)))$

Как и в случае различия точек, различие прямых усматривается специальным проверочным оператором "разныепрямые". Для удобства компиляции, в теоремах приемов вместо терма "$\neg$(прямая(AB) = прямая(AC))" используется терм "разныепрямые(прямая(AB), прямая(AC))". Все антецеденты приема обрабатываются проверочными операторами. Уровень срабатывания равен 3.

(c) Простейшие неравенства для углов.

$\forall_{ABCD}$(биссектриса$(ABCD) \rightarrow 0 < \pi - 2\angle(ABD))$

Утверждение "биссектриса$(ABCD)$" означает, что точка D лежит на биссектрисе невырожденного угла $\angle(ABC)$. Величина этого невырожденного угла, равная $2\angle(ABD)$, меньше π.

$\forall_{ABCab}(a = \angle(BAC) \ \& \ b = \angle(ABC)$ & разныепрямые(прямая(AB), прямая(AC)) & разныеточки$(A, C) \rightarrow 0 < \pi - a - b)$

Первый и второй антецеденты берутся в посылках, третий и четвертый - обрабатываются проверочными операторами. Неравенство означает, что сумма двух внутренних углов треугольника меньше π.

$\forall_{ABCamn}(a = \angle(ABC) \ \& \ 0 < m - n \rightarrow 0 < m\pi - an)$

Переменные m, n идентифицируются с константными выражениями, a - с переменной. Первый антецедент берется в посылках, второй обрабатывается проверочным оператором. Уровень срабатывания приведенных приемов равен 3.

$\forall_{ABCa}(a = \angle(BAC)$ & разныепрямые(прямая(AB), прямая(AC)) $\rightarrow 0 < a)$

Усматривается положительность величины невырожденного угла. Уровень срабатывания равен 4.

(d) Усмотрение острого либо тупого угла. Основная часть приемов, усматривающих острые либо тупые углы, находится в проверочном операторе "усмменьшеилиравно" для нестрогих неравенств. В случае строгих неравенств таких приемов совсем немного. Прежде всего, усмотрение острого угла может быть основано на том факте, что он относится к прямоугольному треугольнику:

$\forall_{ABCa}(a = \angle(ABC)\ \&\ \text{прямая}(AB)\perp\text{прямая}(BC) \to 0 < \pi - 2a)$

Первый антецедент берется в посылках, второй - выделен указателем "усм", т.е. обрабатывается идентифицирующими операторами. О них уже говорилось при общем описании языка ГЕНОЛОГ и будет подробнее рассказано в разделах, посвященных элементарной геометрии. Если равенство для величины угла в посылках отсутствует, применяется другая версия приема:

$\forall_{ABC}(\text{прямая}(AB)\perp\text{прямая}(BC) \to 0 < \pi - 2\angle(BAC))$

Уровни срабатывания равны 4. Несколько простых приемов используют неравенства для угла, уже имеющиеся в посылках:

$\forall_{ABCa}(a = \angle(ABC)\ \&\ \angle(ABC) < \pi/2 \to 0 < \pi/2 - a)$

$\forall_{ABC}(\angle(ABC) < \pi/4 \to 0 < \pi/2 - 2\angle(ABC))$

$\forall_{ABC}(\pi/4 < \angle(ABC) \to 0 < 2\angle(ABC) - \pi/2)$

Первый прием срабатывает на уровне 4, причем выражение a не должно иметь заголовка "угол". Остальные приемы срабатывают на уровне 3. Следующий прием усматривает острый вписанный угол, одна из сторон которого является диаметром окружности:

$\forall_{ABCDE}(C \in \text{окружность}(AB)\ \&\ D \in \text{окружность}(AB)\ \&\ A \in \text{прямая}(DE)$ $\&\ \text{точкалуча}(D, A, E)\ \&\ \text{разныеточки}(C, D) \to 0 < \pi/2 - \angle(CDE))$

Первые четыре антецедента обрабатываются идентифицирующими операторами, пятый - проверочным оператором. Уровень срабатывания приема равен 3. Наконец, имеется прием, усматривающий острый угол треугольника, лежащий против стороны, не являющейся наибольшей:

$\forall_{ABC}(l(AB) = a\ \&\ l(BC) = b\ \&\ 0 \le a - b\ \&\ \text{разныеточки}(A, B)\ \&\ \text{разныеточки}(A, C)\ \&\ \text{разныепрямые}(\text{прямая}(AB), \text{прямая}(AC)) \to 0 < \pi/2 - \angle(BAC))$

Первые два антецедента берутся в посылках. Проверяется, что определяемые ими выражения a, b для длин сторон треугольника не содержат символов "расстояние", "угол". Оставшиеся антецеденты обрабатываются проверочными операторами. Прием имеет сильный ограничитель трудоемкости. Уровень срабатывания равен 3.

(e) Усмотрение невырожденного угла.

$\forall_{ABC}(\neg(B \in \text{прямая}(AC)) \to 0 < \angle(BAC))$

$\forall_{ABC}(\neg(B \in \text{прямая}(AC)) \to 0 < \pi - \angle(BAC))$

Первый антецедент обрабатывается проверочным оператором. Уровень срабатывания равен 3. Если выражение a равно величине некоторого угла, то можно попытаться использовать проверочный оператор, заменив это выражение на величину угла:

$\forall_{ABCa}(\angle(ABC) = a\ \&\ 0 < \pi - \angle(ABC) \to 0 < \pi - a)$

Первый антецедент берется из посылок, второй - обрабатывается проверочным оператором. Уровень срабатывания равен 3. Еще два приема усматривают невырожденность угла четырехугольника:

$\forall_{ABCDa}(\text{четырехугольник}(ABCD)\ \&\ \angle(BAD) = a \to 0 < a)$

$\forall_{ABCDa}(\text{четырехугольник}(ABCD)\ \&\ \angle(BAD) = a \to 0 < \pi - a)$

Указатель "вариант(фикс(1)ромб параллелограмм трапеция)" разрешает варьировать заголовок "четырехугольник" первого антецедента. Указатель "циклупорядочение(фикс(1))" разрешает выполнять при идентификации произвольные циклические перестановки вершин четырехугольника. Уровень срабатывания равен 3.

(f) Трехчлен из теоремы косинусов. Для усмотрения положительности возникающего при использовании теоремы косинусов квадратного трехчлена создан следующий прием:

$\forall_{abcd}(\neg(a - b = 0) \to 0 < a^2 + b^2 + 2abd\cos c)$

Коэффициент d идентифицируется с ±1. Первый антецедент обрабатывается проверочным оператором. Он означает различие длин двух сторон треугольника. В этом случае квадрат длины третьей стороны, выражаемый трехчленом, будет ненулевым. Введен сильный ограничитель трудемкости. Уровень срабатывания равен 1.

(g) Длина вектора.

$\forall_{abc}(\neg(\text{компланарны}(a, b, c)) \to 0 < \text{ длина}(a))$

Антецедент берется в посылках. Заметим, что в силу симметричности операндов отношения компланарности переменная a будет идентифицирована с произвольным из этих операндов. Уровень срабатывания равен 3.

$\forall_{aKi}(\neg(\text{крд}(a, K, i) = 0) \to 0 < \text{длина}(a))$

Напомним, что "крд(a, K, i)" обозначает i-ю координату вектора a в системе координат K. Указатель "контекст(посылка(x2)позиция(x3 x2)вид(x3 крд(x1 x36 x9)))" определяет усмотрение в некоторой посылке подвыражения "крд(a, K, i)", после чего проверочный оператор устанавливает отличие значения этого выражения от нуля. Уровень срабатывания равен 3.

$\forall_a(\text{Вектор}(a)\ \&\ \neg(a = \text{вектор0}) \to 0 < \text{длина}(a))$

Второй антецедент берется в посылках, первый - обрабатывается проверочным оператором. Уровень срабатывания равен 2.

25. Конечные сумма и произведение.

Чтобы проверить положительность конечной суммы либо конечного произведения, предпринимается попытка усмотреть положительность общего члена:

$$\forall_{fg}(0 < f(x) \to 0 < \prod_{x,g(x)} f(x))$$

$$\forall_{fg}(0 < f(x) \to 0 < \sum_{x,g(x)} f(x))$$

Антецедент обрабатывается проверочным оператором, причем указатель "занесениепосылки(1 значение(x7 x23))" передает этому оператору дополнительную информацию $g(x)$ о параметре x. Переменные f, g выделены указателем "отображение". Уровень срабатывания равен 1.

26. Символы бесконечности.

    $\forall_a(a - \text{число} \rightarrow a < \infty)$

    $\forall_a(a - \text{число} \rightarrow -\infty < a)$

    Антецедент обрабатывается проверочным оператором. Уровень срабатывания равен 1.

27. Учет предельных условий. Если в посылках имеется утверждение $(x \rightarrow a\backslash b)$ о "стремлении" переменной x к a с заданным условием b на окрестность точки a, то для усмотрения строгих неравенств используются следующие дополнительные приемы:

    (a) Определение строгого знака предела.

    $\forall_{abx}((x \rightarrow a\backslash b)\ \&\ 0 < a \rightarrow 0 < x)$

    $\forall_{abx}((x \rightarrow a + 0)\ \&\ 0 \leq a \rightarrow 0 < x)$

    Второй антецедент обрабатывается проверочным оператором. Второй прием предполагает, что в тех контекстах, где встречается посылка о стремлении x к a справа, всегда будет можно считать x отличным от a. Напомним, что посылка о "стремлении" означает условное соглашение, позволяющее избежать громоздких кванторных записей, так что нужно лишь проследить корректность в указанном смысле всех используемых кванторных импликаций. Уровень срабатывания равен 1. Следующие приемы обобщают проверку на случай линейных относительно x выражений:

    $\forall_{abcx}((x \rightarrow c + 0)\ \&\ 0 < a\ \&\ 0 \leq ac + b \rightarrow 0 < ax + b)$

    $\forall_{abcx}((x \rightarrow c - 0)\ \&\ a < 0\ \&\ 0 \leq ac + b \rightarrow 0 < ax + b)$

    $\forall_{abcx}((x \rightarrow c - 0)\ \&\ 0 < a\ \&\ ac + b \leq 0 \rightarrow ax + b < 0)$

    Проверяется, что x не входит в a, b. Второй и третий антецеденты обрабатываются проверочными операторами. Уровень срабатывания равен 2. Для случая произвольной непрерывной зависимости от x созданы еще два приема:

    $\forall_{abcx}((x \rightarrow a\backslash b)\ \&\ c = \lim_{x\rightarrow a\backslash b} f(x)\ \&\ (c = \infty\ \vee\ 0 < c) \rightarrow 0 < f(x))$

    $\forall_{abcx}((x \rightarrow a\backslash b)\ \&\ c = \lim_{x\rightarrow a\backslash b} f(x)\ \&\ (c = -\infty\ \vee\ c < 0) \rightarrow f(x) < 0)$

    $f(x)$ - произвольное выражение, содержащее x. Должна отсутствовать посылка "комплексное(x)". Второй антецедент выделен указателем "идентификатор". Он обращается к нормализатору "нормпредел", вычисляющему предел c выражения $f(x)$ в точке a. Третий антецедент выделен указателем "блокпроверок". Проверяется, что c есть либо символ бесконечности, либо число соответствующего знака. В обоих случаях нормализатор "нормпредел" сумел вычислить предел, и значит, функция $f(x)$ в рассматриваемой точке была непрерывна. Уровень срабатывания равен 3. Созданы также два аналогичных приема, относящихся к случаю, когда $f(x)$ - вещественнозначная функция комплексного переменного x. Их теоремы отличаются от вышеприведенных лишь наличием дополнительного антецедента "комплексное(x)". Вместо вещественнозначного нормализатора "номпредел" здесь используется комплекснозначный нормализатор "нормПредел". Уровень срабатывания равен 3.

(b) Использование знака производной при нулевом пределе функции.

$$\forall_{afx}(x \to a+0 \ \& \ \lim_{x\to a+0} f(x) = 0 \ \& \ \lim_{x\to a+0}\frac{df(x)}{dx} < 0 \to f(x) < 0)$$

$$\forall_{afx}(x \to a+0 \ \& \ \lim_{x\to a+0} f(x) = 0 \ \& \ 0 < \lim_{x\to a+0}\frac{df(x)}{dx} \to 0 < f(x))$$

$$\forall_{afx}(x \to a-0 \ \& \ \lim_{x\to a-0} f(x) = 0 \ \& \ \lim_{x\to a-0}\frac{df(x)}{dx} < 0 \to f(x) < 0)$$

$$\forall_{afx}(x \to a-0 \ \& \ \lim_{x\to a-0} f(x) = 0 \ \& \ 0 < \lim_{x\to a-0}\frac{df(x)}{dx} \to 0 < f(x))$$

Первый антецедент берется в посылках, второй выделен указателем "идентификатор". Он вычисляет предел с помощью пакетного нормализатора "нормпредел". Третий антецедент обрабатывается проверочным оператором. Производная вычисляется нормализатором "нормпроизводная" и упрощается вспомогательной задачей на преобразование, решаемой до максимального уровня 4. Затем для вычисления предела используется нормализатор "нормпредел". Введен сравнительно сильный ограничитель трудоемкости. Уровень срабатывания равен 3.

(c) Использование асимптотической оценки при нулевом пределе функции.

$\forall_{abfg}(x \to a\backslash b \ \& \ \lim_{x\to a\backslash b} f(x) = 0 \ \& \ g = \lambda_x(f(x), x - \text{число}) \ \& \ 0 < g(x) \to 0 < f(x))$

$\forall_{abfg}(x \to a\backslash b \ \& \ \lim_{x\to a\backslash b} f(x) = 0 \ \& \ g = \lambda_x(f(x), x - \text{число}) \ \& \ g(x) < 0 \to f(x) < 0)$

Первые два антецедента обрабатываются так же, как в предыдущем пункте. Затем к $f(x)$ применяется нормализатор "асимптоценка", определяющий асимптотическую оценку g в предположении о стремлении x к $a\backslash b$. Последний антецедент предпринимает попытку определить строгий знак $g(x)$ с помощью проверочного оператора. Уровень срабатывания равен 4.

28. Тригонометрические функции.

(a) Неравенства с синусом.

$\forall_a(0 < a \ \& \ 0 < \pi - a \to 0 < \sin a)$

$\forall_a(0 < a - \pi \ \& \ 0 < 2\pi - a \to \sin a < 0)$

Антецеденты обрабатываются проверочным оператором. Либо в посылках упоминается угол, либо имеется неравенство, пересекающееся по своим параметрам с a и не содержащее тригонометрических операций. Второй прием имеет сильный ограничитель трудоемкости. Уровень срабатывания равен 3.

$\forall_{abc}(0 \leq b - 1 \ \& \ 0 < c - 1 \to 0 < c - (\sin a)^b)$

$\forall_{abc}(0 \leq b-1 \;\&\; 0 < c-1 \to 0 < c + (\sin a)^b)$

Переменные b, c идентифицируются с константами. Антецеденты обрабатываются проверочными операторами. Уровень срабатывания равен 1.

$\forall_a(0 \leq \sin a \;\&\; 0 \leq \cos a \to 0 < \sin(a + \pi/4))$

Оба антецедента берутся в посылках, причем в каждом случае допускается как нестрогий, так и строгий знак неравенства. Уровень срабатывания равен 1.

(b) Неравенства с косинусом. Приемы аналогичны приемам для синуса:

$\forall_a(0 \leq a \;\&\; 0 < \pi/2 - a \to 0 < \cos a)$

$\forall_{abc}(0 \leq b-1 \;\&\; 0 < c-1 \to 0 < c + (\cos a)^b)$

$\forall_{abc}(0 \leq b-1 \;\&\; 0 < c-1 \to 0 < c - (\cos a)^b)$

(c) Связь знаков тангенса и котангенса.

$\forall_a(0 < \operatorname{ctg} a \to 0 < \operatorname{tg} a)$

$\forall_a(0 < \operatorname{tg} a \to 0 < \operatorname{ctg} a)$

Антецедент берется в посылках. Уровень срабатывания равен 2.

(d) Сумма специального вида.

$\forall_{abc}(0 < a \;\&\; 0 < b \to 0 < a\cos c - b\cos c + a + b)$

Антецеденты обрабатываются проверочными операторами. Вместо косинуса может идентифицироваться синус. Уровень срабатывания равен 3.

29. Обратные тригонометрические функции.

(a) Неравенства с арксинусом.

$\forall_a(0 < a \to 0 < \arcsin a)$

$\forall_a(a < 0 \to \arcsin a < 0)$

Антецедент обрабатывается проверочным оператором, уровень срабатывания равен 2.

$\forall_a(0 < \pi - \arcsin a)$

$\forall_a(0 < \pi + \arcsin a)$

Уровень срабатывания равен 3.

(b) Неравенства с арккосинусом.

$\forall_{ab}(0 < a - \pi \to \arccos b < a)$

$\forall_a(0 < a + 1 \to 0 < \pi - \arccos a)$

Антецедент обрабатывается проверочным оператором. Уровень срабатывания равен 2.

(c) Неравенства с арктангенсом.

$\forall_a(a < 0 \to \operatorname{arctg} a < 0)$

$\forall_a(0 < a \to 0 < \operatorname{arctg} a)$

$\forall_{ab}(0 < 2b - \pi \to \operatorname{arctg} a < b)$

$\forall_{ab}(2b + \pi < 0 \to b < \operatorname{arctg} a)$

$\forall_{ab}(2b + \pi < 0 \to \operatorname{arctg} a + b < 0)$

$\forall_{ab}(0 < 2b - \pi \to 0 < \operatorname{arctg} a + b)$

Антецедент обрабатывается проверочным оператором. Уровень срабатывания равен 2.

$\forall_a(0 < \pi/2 - \operatorname{arctg} a)$

$\forall_a(0 < \pi/2 + \operatorname{arctg} a)$

Уровень срабатывания равен 3.

30. Целочисленные выражения.

    (a) Натуральное число.

    $\forall_a(a - \text{натуральное} \to 0 < a)$

    Антецедент берется в посылках. Уровень срабатывания равен 3.

    $\forall_{mn}(0 < \text{нод}(m, n))$

    Согласно о.д.з. для "нод", числа m, n должны быть целыми и отличными от 0. Поэтому дополнительная проверка отличия от нуля не выполняется. Уровень срабатывания равен 1.

    (b) Целая часть.

    $\forall_{abcdn}(n - \text{целое} \ \& \ 0 < d \ \& \ ac = b \to 0 < an - b[n/c] + d)$

    Переменные a, b, c идентифицируются с натуральными константами. Указатель "единица" разрешает им обращаться в единицу. Первые два антецедента обрабатываются проверочными операторами, третий - выделен указателем "программа". Уровень срабатывания равен 4.

    $\forall_a(a < 0 \to [a] < 0)$

    Антецедент обрабатывается проверочным оператором. Уровень срабатывания равен 2.

    (c) Факториал.

    $\forall_n(0 < n!)$

    Уровень срабатывания равен 1.

    (d) Положительное целое число не меньше единицы.

    $\forall_{abn}(n - \text{целое} \ \& \ 0 < n \ \& \ 0 \leq a \ \& \ 0 < a + b \to 0 < an + b)$

    Первые два антецедента берутся в посылках, третий и четвертый обрабатываются проверочными операторами. Коэффициент a может обращаться в единицу. Уровень срабатывания равен 3.

    (e) Длина непустого набора.

    $\forall_a(\neg(a = \text{пустоеслово}) \to 0 < l(a))$

    Выражение $l(a)$ в скобочной записи имеет вид "длинанабора(a)". Антецедент обрабатывается проверочным оператором. Уровень срабатывания равен 2.

    (f) Использование принадлежности конечному отрезку целых чисел.

    $\forall_{ijmn}(j \in \{m, \ldots, n\} \ \& \ 0 < i - n \to 0 < i - j)$

    Первый антецедент берется в посылках, второй обрабатывается проверочным оператором. Имеется сильный ограничитель трудоемкости. Уровень срабатывания равен 2.

    $\forall_{ijmn}(j \in \{m, \ldots, n\} \ \& \ 0 < m + i \to 0 < j + i)$

    Антецеденты обрабатываются так же, как в предыдущем приеме. Уровень срабатывания равен 3.

(g) Число сочетаний.

$\forall_{nk}(k -$ целое & $n -$ целое $\&0 \leq k$ & $0 \leq n-k \rightarrow 0 < C_n^k)$

Все антецеденты обрабатываются проверочными операторами. Уровень срабатывания равен 3.

(h) Простое число.

$\forall_a($простое$(a) \rightarrow 0 < a-1)$

Антецедент обрабатывается проверочным оператором, уровень срабатывания равен 2.

$\forall_a($простое$(a) \rightarrow 0 < a)$

Антецедент берется в посылках, уровень срабатывания равен 3.

(i) Дробная часть.

$\forall_a(0 < 1 -$ дробнаячасть$(a))$

Уровень срабатывания равен 3.

31. Гиперболические функции.

$\forall_a(0 < \operatorname{ch} a)$

$\forall_a(0 < a \rightarrow 0 < \operatorname{cth} a)$

$\forall_a(0 < a \rightarrow 0 < \operatorname{th} a)$

$\forall_a(0 < a \rightarrow 0 < \operatorname{sh} a)$

$\forall_a(a < 0 \rightarrow \operatorname{sh} a < 0)$

Антецеденты обрабатываются проверочными операторами. В двух последних приемах испоьзуется указатель "спуск", блокирующий альтернативные попытки. Уровень срабатывания первых четырех приемов равен 1, последнего приема - 2.

32. Сумма с экспонентой.

$\forall_{abcd}(1 < b$ & $0 < a$ & $0 \leq a+d$ & $0 < c \rightarrow 0 < ab^c + d)$

Переменные a, b, d идентифицируются с константами, c - с неконстантным выражением. Антецеденты обрабатываются проверочными операторами. Уровень срабатывания равен 1.

33. Мощность непустого множества.

$\forall_A(\neg(A = \emptyset) \rightarrow 0 < \mathrm{card} A)$

Антецедент обрабатывается проверочным оператором. Чтобы заблокировать встречную попытку усмотреть непустоту множества через оценку его мощности, обращение к оператору сопровождается комментарием "усмне0". Уровень срабатывания равен 2.

34. Физические величины. Представлены достаточно разрозненные сведения о положительности величин, оказавшиеся нужными для проработанных задач по элементарной физике. Во многих случаях величина считается положительной по умолчанию, в отсутствии явных указаний противного. Необходимые фильтры, контролирующие наличие таких указаний, вставлены лишь эпизодически, и по мере надобности будут дополняться.

(a) Единицы измерения. Единицы измерения оказалось удобным обрабатывать так, как если бы они были числами. Это всего лишь технический прием, упрощающий программирование. Так как никаких дополнительных сведений об этих числах (кроме их положительности и взаимной связи) не используется, то все выкладки оказываются корректными.

$0 <$ дн.

Здесь "дн." - условная единица измерения "день", встречающаяся в текстовых арифметических задачах.

$0 <$ час, $0 <$ км, $0 <$ мин, $0 <$ сек, $0 <$ л, $0 <$ руб, $0 <$ м, $0 <$ т, $0 <$ г, $0 <$ кг, $0 <$ год, $0 <$ см, $0 <$ Н, $0 <$ Дж, $0 <$ кВт, $0 <$ моль

Уровень срабатывания равен 1.

(b) Средняя скорость при невырожденном пути.

$\forall_{apT}(\text{Путь}(a,T) = p \ \& \ \neg(\text{длина}(p) = 0) \to 0 < \text{скорость}(a,T))$

Уровень срабатывания равен 3.

(c) Площадь сечения русла.

$\forall_{aART}(R = \text{русло}(a,T) \to 0 < S(\text{сечениерусла}(R,A)))$

Здесь "русло(a,T)" - русло течения a на промежутке времени T, "сечениерусла(R,A)" - сечение русла R в точке A его направляющей линии. Уровень срабатывания равен 2.

(d) Радиус внешней окружности круглого тела.

$\forall_a(0 < \text{радиус}(\text{внешокружность}(a)))$

Уровень срабатывания равен 2.

(e) Абсолютная температура.

$\forall_{at}(0 < \text{абстемпература}(a,t))$

Уровень срабатывания равен 1.

(f) Давление.

$\forall_{at}(0 < \text{давление}(a,t))$

Уровень срабатывания равен 1.

(g) Константы.

$0 <$ газконст, $0 <$ числоАвогадро, $0 <$ радиус(Земля), $0 <$ грав.

Уровень срабатывания равен 1.

(h) Масса.

$\forall_a(0 < \text{масса}(a))$

Проверяется отсутствие посылки, явно указывающей, что масса a равна 0. Уровень срабатывания равен 1.

(i) Количество вещества.

$\forall_{ab}(\text{вещество}(a,b) \ \& \ 0 < \text{масса}(a) \to 0 < \text{количествовещества}(a,b))$

Утверждение "вещество(a,b)" означает, что объект a состоит из вещества b. Второй антецедент обрабатывается проверочным оператором. Уровень срабатывания равен 2.

35. Монотонность логарифмической функции.

$\forall_{axy}(x < y \ \& \ 0 < a - 1 \to \log_a x < \log_a y)$

$\forall_{axy}(y < x \;\&\; 0 < -a+1 \to \log_a x < \log_a y)$

$\forall_{abc}(0 < a-1 \;\&\; 0 < c-b \to 0 < \log_a c - \log_a b)$

$\forall_{abc}(0 < a-1 \;\&\; c-b < 0 \to \log_a c - \log_a b < 0)$

Антецеденты обрабатываются проверочными операторами. Уровень срабатывания равен 2.

36. Функция распределения случайной величины. Значение ее не превосходит единицы:

    $\forall_{abfgxP}(0 \leq b \;\&\; 0 < a-b \;\&\; f = \text{функраспред}(g,P) \to 0 < a - bf(x))$

    Первые два антецедента обрабатываются проверочными операторами, третий - берется в посылках. Уровень срабатывания равен 4.

37. Значение перестановки.

    $\forall_{fmnij}(\text{перестановка}(f,\{m,\ldots,n\}) \;\&\; f(i)=m \;\&\; \neg(j-i=0) \to 0 < f(j)-m)$

    $\forall_{fmnij}(\text{перестановка}(f,\{m,\ldots,n\}) \;\&\; f(i)=n \;\&\; \neg(j-i=0) \to 0 < n-f(j))$

    Первые два антецедента берутся в посылках, третий - обрабатывается проверочным оператором. Уровень срабатывания равен 3.

Проверочный оператор "провменьше"

Оператор "провменьше" представляет собой надстройку над оператором "усмменьше". Он обращается к последнему, и в случае неудачи предпринимает попытки воспользоваться рядом дополнительных приемов. В основном, эти приемы выполняют различные группировки и оценки, использующие неравенство для среднего арифметического и среднего геометрического. Чтобы обратиться из какого-либо приема к оператору "провменьше", нужно соответствующий антецедент выделить не указателем "блокпроверок", а указателем "проверка". Оператор имеет следующие приемы:

1. Обращение к оператору "усмменьше".

   $\forall_{ab}(a < b \to a < b)$

   Антецедент выделен указателем "блокпроверок" и обрабатывается оператором "усмменьше". Уровень срабатывания равен 1.

2. Поглощение одного слагаемого двумя с применением квадратичной оценки.

   $\forall_{abcdefghi}(0 \leq a \;\&\; 0 \leq b \;\&\; c^2 = ab \;\&\; 0 < d \;\&\; 0 < e \;\&\; f = 4de - g^2 \;\&\; 0 \leq f \;\&\; 4di < bf + 4dh \to i < da + eb + gc + h)$

   Коэффициенты d, e, g идентифицируются с десятичными константами. Проверяется, что первые две из них положительны, знак последней несущественен. Выражения a, b должны быть неотрицательны, а квадрат выражения c должен быть равен произведению ab. Затем предпринимается попытка так разбить коэффициент e на две части m, n, $m > 0$, чтобы выполнялось неравенство $ad + mb + gc \geq 0$, т.е. слагаемое cg (как удвоенное среднее геометрическое) оказалось поглощено суммой $ad + mb$ (удвоенным средним арифметическим). Очевидно, для этого достаточно выполнение условия $g^2c^2 = 4admb$, или $m = g^2/4d$. После отбрасывания в правой части проверяемого неравенства $i < da + eb + gc$ неотрицательной суммы получаем неравенство $i < (e - g^2/4d)b + h$. Вычисляя вспомогательное значение $f = 4de - g^2$ и избавляясь от знаменателя, приводим последнее к виду $4di < fb + 4dh$. Его снова обрабатываем оператором "провменьше", предварительно раскрыв скобки в правой части. Первые два антецедента обрабатываются оператором "усмменьшеилиравно", третий - выделен указателем "идентификатор". Антецеденты с четвертого по седьмой выделены указателем "программа". Наконец, последний антецедент выделен указателем "проверка". Если очевидна неотрицательность слагаемого gc, то прием блокируется. Он блокируется также, если число слагаемых правой части больше четырех. Очередность идентификации регулируется указателями "определено(фикс(0 2 2)x2)", "определено(фикс(0 2 1)x3)". Они означают, что перед выбором слагаемого eb уже должно быть определено выражение b, а перед выбором слагаемого da - выражение c. Таким образом, сначала выбирается слагаемое gc, затем - слагаемое da, затем с помощью третьего антецедента определяется b, и лишь затем берется подходящее слагаемое eb. Уровень срабатывания равен 3.

$\forall_{abcdefpqrh}(a = bd \ \& \ 0 \leq d \ \& \ 0 < q \ \& \ e = 2p - q \ \& \ f = 2r - q \ \& \ 0 \leq e \ \& \ 0 \leq f \ \& \ 0 < eab + fdc^2 + 2h \rightarrow 0 < pab + qac + rdc^2 + h)$

$\forall_{abcdefpqrh}(a = bd \ \& \ 0 \leq d \ \& \ q < 0 \ \& \ e = 2p + q \ \& \ f = 2r + q \ \& \ 0 \leq e \ \& \ 0 \leq f \ \& \ 0 < eab + fdc^2 + 2h \rightarrow 0 < pab + qac + rdc^2 + h)$

В этих приемах неотрицательные члены pab, rdc^2 поглощают член qac. Если $0 < q$ (первый прием), то, с учетом первого антецедента, получаем, что среднее арифметическое выражений qab, qdc^2 не менее их среднего геометрического qac. В этом случае вычитаем из коэффициентов p, r необходимое для поглощения значение $q/2$, отбрасываем поглощенный член, и после домножения на 2 приходим к последнему антецеденту, снова обрабатываемому оператором "провменьше". Второй прием аналогичным образом рассматривает случай $q < 0$. Порядок идентификации в обоих приемах таков: выбираются слагаемые pab, qac; коэффициенты p, q определяются как произведения всех десятичных констант, являющихся сомножителями этих слагаемых; выражение a определяется как произведение оставшихся их общих множителей. Таким образом, находятся b, c. Далее первый антецедент, выделенный указателем "идентификатор", определяет выражение d. Наконец, выбирается третье слагаемое rdc^2 и определяется r. Проверяется, что коэффициенты p, q, r суть десятичные константы. Второй антецедент обрабатывается проверочным оператором, антецеденты с третьего по седьмой выделены указателем "программа". Если очевидна неотрицательность поглощаемого члена либо отсутствуют кратные вхождения переменных в проверяемое неравенство, то прием блокируется. Блокировка происходит также, если лишь для одного слагаемого правой части неочевидна его неотрицательность, а общее число слагаемых более 5. Если удается усмотреть, что $d \neq 0, b \neq c$, то в последнем антецеденте берется нестрогое неравенство. При этом вместо оператора "провменьше" применяется оператор "провменьшеилиравно". Данные действия обеспечиваются указателем "замещение(...)". Уровень срабатывания равен 1.

3. Поглощение части одного слагаемого двумя слагаемыми с применением квадратичной оценки.

$\forall_{abcdepqrh}(a = bd \ \& \ 0 \leq d \ \& \ 0 < q \ \& \ pr = e^2 \ \& \ 0 < p \ \& \ 0 < r \ \& \ 0 < (q - 2e)ac + h \rightarrow 0 < pab + qac + rdc^2 + h)$

$\forall_{abcdepqrh}(a = bd \,\&\, 0 \le d \,\&\, q < 0 \,\&\, pr = e^2 \,\&\, 0 < p \,\&\, 0 < r \,\&\, 0 < (q+2e)ac+h \to 0 < pab + qac + rdc^2 + h)$

Здесь неотрицательные слагаемые pab, rdc^2 целиком используются для поглощения "части" слагаемого qac. Их удвоенное среднее геометрическое равно $2\sqrt{pr}ac$. Прием применяется лишь в том случае, когда pr является точным квадратом некоторой целочисленной константы e. Тогда поглощенной оказывается часть $2eac$, а остаток $(q - 2e)ac$ сохраняется в последнем антецеденте, обрабатываемом рекурсивным обращением к тому же оператору. Уровень срабатывания равен 2.

4. Поглощение одного слагаемого двумя с двукратным использованием одного из них.

   $\forall_{abcdefghijk}(0 \le a \,\&\, 0 \le b \,\&\, c^2 = ab \,\&\, d^2 = ac \,\&\, e = 4f - 3|g| \,\&\, 0 \le e \,\&\, h = 4i - |g| \,\&\, 0 \le h \,\&\, 4k < ea + hb + 4j \to k < fa + ib + gd + j)$

   Прием поглощает слагаемое gd суммой слагаемых fa, ib, используя неравенства $|g|a/4 + |g|b/4 \ge |g|c/2$ и $|g|a/2 + |g|c/2 \ge gd$. Предполагается, что ab есть полный квадрат некоторого выражения c, причем ac равно d^2. После применения указанных неравенств слагаемое gd отбрасывается, а от поглощающих членов остаются слагаемые $(f - 3|g|/4)a, (i - |g|/4)b$. Домножая обе части полученного неравенства на 4, приходим к последнему антецеденту, обрабатываемому тем же оператором "провменьше". Идентификация происходит в следующем порядке. Сначала выбирается слагаемое gd, причем проверяется, что не очевидна его неотрицательность. Переменная g идентифицируется с численным коэффициентом, d - с остальными множителями. Затем идентифицируется слагаемое fa, в котором выделяются численный коэффициент f и остаток a. Четвертый антецедент, выделенный указателем "идентификатор", определяет выражение c, а третий - выражение b, Далее идентифицируется ib и определяется численный коэффициент i. Первый и второй антецеденты обрабатываются оператором "усмменьшеилиравно", антецеденты с пятого по восьмой выделены указателем "программа". Уровень срабатывания равен 2.

5. Поглощение одного слагаемого тремя с применением квадратичной оценки.

   $\forall_{abcdefghijklmn}(0 \le a \,\&\, 0 \le b \,\&\, 0 \le c \,\&\, d^2 = bc \,\&\, e^2 = ad \,\&\, f = 2g - |h| \,\&\, 0 \le f \,\&\, i = 4j - |h| \,\&\, 0 \le i \,\&\, k = 4l - |h| \,\&\, 0 \le k \,\&\, 4n < 2fa + ib + kc + 4m \to n < ga + jb + lc + he + m)$

   Прием поглощает слагаемое he суммой слагаемых ga, jb, lc, используя для этого неравенства $|h|b/4 + |h|c/4 \ge |h|d/2$, $|h|d/2 + |h|a/2 \ge |h|e$. Предполагается, что bc есть полный квадрат некоторого выражения d, причем ad равно e^2. После отбрасывания поглощенного и поглощающих слагаемых сохраняются остатки поглощающих членов $(g - |h|/2)a, (j - |h|/4)b, ((l - |h|/4)c$. Домножая обе части полученного неравенства на 4, приходим к последнему антецеденту теоремы приема. Уровень срабатывания равен 3.

6. Поглощение одно слагаемого тремя с применением кубической оценки.

   $\forall_{abcdefghijklmn}(0 \le a \,\&\, 0 \le b \,\&\, 0 \le c \,\&\, d^3 = an \,\&\, n = bc \,\&\, e = 3f - |g| \,\&\, 0 \le e \,\&\, h = 3i - |g| \,\&\, 0 \le h \,\&\, j = 3k - |g| \,\&\, 0 \le j \,\&\, 3m < ea + hb + jc + 3l \to fa + ib + kc + gd + l)$

Прием поглощает слагаемое gd суммой слагаемых fa, ib, kc, используя для этого неравенство $|g|a/3+|g|b/3+|g|c/3 \geq gd$. Предполагается, что $abc = d^3$. Порядок идентификации таков. Сначала выбирается слагаемое gd, причем проверяется, что не очевидна его неотрицательность. По этом слагаемому находятся числовой коэффициент g и произведение d остальных множителей. Затем выбирается слагаемое fa, определяющее f, a. Четвертый антецедент позволяет теперь идентифицировать n. Далее выбирается слагаемое ib, по которому определяются i, b. Пятый антецедент позволяет идентифицировать c, и, наконец, выбирается слагаемое kc. Уровень срабатывания равен 4.

7. Поглощение части одного слагаемого тремя слагаемыми.

   $\forall_{abcdefghijkl}(0 \leq a \ \& \ 0 \leq b \ \& \ 0 \leq c \ \& \ d^3 = al \ \& \ l = bc \ \& \ e^3 = fgh \ \& \ 0 < f \ \& \ 0 < g \ \& \ 0 < h \ \& \ k < (i+3e)d + j \rightarrow k < fa + gb + hc + id + j)$

   Прием поглощает часть $-3ed$ слагаемого id, используя неравенство $fa+gb+hc \geq 3\sqrt[3]{\overline{fgh}}d$. Предполагается, что $d^3 = abc, 0 < f, 0 < g, 0 < h$, причем fgh является полным кубом некоторой целочисленной константы e. Уровень срабатывания равен 5.

8. Поглощение одного слагаемого двумя с применением кубической оценки.

   $\forall_{abcdefghij}(0 \leq a \ \& \ 0 \leq b \ \& \ c^3 = a^2b \ \& \ d = 2e - f \ \& \ 0 \leq d \ \& \ g = 2h + 3f \ \& \ 0 \leq g \ \& \ 2j < db + gc + 2i \rightarrow j < fa + eb + hc + i)$

   Используется неравенство $fa+fb+fb \geq 3fc$. Предполагается, что $c^3 = a^2b, 0 \leq a, 0 \leq b$, причем не усматривается неотрицательность hc. Тогда $h \leq 0$, и из условий $g = 2h+3f, 0 \leq g$ следует неотрицательность f. Указанное неравенство для средних арифметического и геометрического преобразуется к виду $0 \leq fb/2 - 3fc/2 + fa$. Если сложить это неравенство и неравенство из последнего антецедента, обе части которого предварительно поделены на 2, то получим неравенство в консеквенте. Уровень срабатывания равен 5.

   $\forall_{abcdefghij}(0 \leq a \ \& \ 0 \leq b \ \& \ c^3 = a^2b \ \& \ d = 3e - 2|f| \ \& \ 0 \leq d \ \& \ j = 3g - |f| \ \& \ 0 \leq j \ \& \ 3i < da + jb + 3h \rightarrow i < ea + gb + fc + h)$

   Для поглощения члена fc используется неравенство $|f|a/3+|f|a/3+|f|b/3 \geq fc$. Уровень срабатывания равен 5.

9. Извлечение нижней оценки произведения из неравенств для сомножителей.

   $\forall_{abcdefgh}(h = \max(dg, ef) \ \& \ 0 \leq d \ \& \ 0 \leq g \ \& \ 0 \leq e \ \& \ 0 \leq f \ \& \ 0 \leq a + d \ \& \ 0 \leq -a + f \ \& \ 0 \leq b + e \ \& \ 0 \leq -b + g \ \& \ 0 < c - h \rightarrow 0 < ab + c)$

   Неравенства $0 \leq a+d \ \& \ 0 \leq -a+f \ \& \ 0 \leq b+e \ \& \ 0 \leq -b+g$ берутся из посылок. Переменные d, e, f, g идентифицируются с десятичными константами, переменные a, b - с неконстантными выражениями. Первые пять антецедентов выделены указателем "программа". Они определяют нижнюю оценку $-h$ произведения ab. Последний антецедент обращается к оператору "провменьше" для проверки неравенства $0 < c - h$. Указатели "альтернатива" разрешают брать в посылках нестрогие версии указанных выше неравенств. Уровень срабатывания равен 2.

10. Извлечение нижней оценки суммы квадратов из неравенства для суммы.

    $\forall_{abcdef}(0 \leq e \ \& \ 0 < ad+bd+c \ \& \ 0 < d \ \& \ 0 \leq ec^2+2d^2f \ \& \ c \leq 0 \rightarrow 0 < a^2e+b^2e+f)$

    $\forall_{abcdef}(0 \leq e \ \& \ 0 \leq ad+bd+c \ \& \ 0 < d \ \& \ 0 < ec^2+2d^2f \ \& \ c \leq 0 \rightarrow 0 < a^2e+b^2e+f)$

Второй антецедент берется в посылках. Если перенести c в левую часть, вынести d за скобку и возвести в квадрат полученное неравенство с неотрицательной левой частью, то будем иметь $c^2 < d^2(a+b)^2$. Так как $(a+b)^2 \leq 2a^2 + 2b^2$, то получаем строгую нижнюю оценку $c^2/2d^2$ для суммы квадратов a^2+b^2. Следовательно, для доказательства неравенства $0 < a^2e+b^2e+f$ достаточно доказать $0 \leq ec^2/2d^2 + f$, или $0 \leq ec^2 + 2d^2f$. Это и есть четвертый антецедент теоремы приема. Он обрабатывается проверочным оператором "провменьшеилиравно". Уровень срабатывания равен 2.

11. Оценка для суммы квадратов через сумму, позволяющая свести правую часть неравенства к виду разности квадратов.

    $\forall_{abcde}(0 \leq e \mathbin{\&} 0 \leq d \mathbin{\&} 0 < \sqrt{d}a + \sqrt{d}b - \sqrt{2}\sqrt{e}c \mathbin{\&} 0 \leq a+b \mathbin{\&} 0 \leq c \rightarrow 0 < a^2d + b^2d - c^2e)$

    Переменные d, e идентифицируются с константными выражениями. третий антецедент обрабатывается оператором "провменьше", прочие - выделены указателем "блокпроверок". Если домножить правую часть доказываемого неравенства на 2, то для суммы $2a^2d + 2b^2d$ будем иметь нижнюю нестрогую оценку $(\sqrt{d}(a+b))^2$. Таким образом, достаточно доказать неравенство $0 < (\sqrt{d}(a+b))^2 - (\sqrt{2}\sqrt{e}c)^2$ для разности квадратов. Ввиду неотрицательности выражений под квадратами, достаточно доказать неравенство $0 < \sqrt{d}a + \sqrt{d}b - \sqrt{2}\sqrt{e}c$. Уровень срабатывания равен 2.

12. Попытка замены суммы квадратов на удвоенное произведение.

    $\forall_{abcdef}(0 \leq a \mathbin{\&} b = d^2 \mathbin{\&} c = e^2 \mathbin{\&} 0 < 2ade+f \mathbin{\&} 0 \leq d \mathbin{\&} 0 \leq e \rightarrow 0 < ab+ac+f)$

    Второй и третий антецеденты выделены указателем "идентификатор". Они усматривают в выражениях b, c полные квадраты выражений d, e. Первый, а также пятый и шестой антецеденты выделены указателем "блокпроверок". Они устанавливают неотрицательность выражений a, d, e. Четвертый антецедент, полученный из доказываемого неравенства заменой суммы $ab + ac = a(d^2 + e^2)$ на $2ade$, обрабатывается оператором "провменьше". Попытка применить прием предпринимается лишь в том случае, когда существуют две различные переменные x, y, входящие одна в ab, а другая в ac, и встречающиеся в f ровно в одном общем для них слагаемом. Количество слагаемых правой части доказываемого неравенства должно быть меньше 6. Указатель "замещение(...)" учитывает возможность замены в четвертом антецеденте строгого неравенства на нестрогое, если $a \neq 0$ и $d \neq e$. Уровень срабатывания равен 5.

13. Попытка замены суммы квадрата и удвоенного произведения на другой квадрат, взятый с обратным знаком.

    $\forall_{abcde}(0 \leq d \mathbin{\&} a = bd \mathbin{\&} 0 < 4e - dc^2 \rightarrow 0 < ab + ac + e)$

    С учетом второго антецедента, доказываемое неравенство имеет вид $0 < b^2d + bdc + e$. Добавив к обеим частям выражение $dc^2/4$ и выделив в правой части полный квадрат, получим неравенство $dc^2/4 < d(b + c/2)^2 + e$. Очевидно, оно является следствием неравенства $dc^2/4 < e$, эквивалентного третьему антецеденту. Первый антецедент выделен указателем "блокпроверок", второй - указателем "идентификатор". Третий антецедент обрабатывается оператором "провменьше". Перед применением приема проверяется существование в посылках неравенства, имеющего общий параметр с d. Проверяется также, что

a, b неконстантные и что число слагаемых правой части неравенства не более трех. Уровень срабатывания равен 5.

14. Оценка сверху для радикала второй степени.

    $\forall_{abc}(a \leq 0 \ \& \ 0 \leq b \ \& \ 0 < ab + a + 2c \rightarrow 0 < a\sqrt{b} + c)$

    Снова используется неравенство для средних арифметического и геометрического: $b/2 + 1/2 \geq \sqrt{b}$, откуда для неположительного a вытекает: $ab/2 + a/2 \leq a\sqrt{b}$. Таким образом, доказываемое неравенство оказывается следствием неравенства $0 < ab/2 + a/2 + c$, т.е. третьего антецедента. Первые два антецедента выделены указателем "блокпроверок", последний - обрабатывается оператором "провменьше". Выражение b должно быть неконстантным, иметь заголовок "плюс" и не являться множителем слагаемых выражения c. Уровень срабатывания равен 5.

15. Оценка, использующая выпуклость степенной функции.

    $\forall_{abcde}(0 < c \ \& \ 0 < 1 - c \ \& \ 0 \leq e \ \& \ 0 < b \ \& \ 0 < d \ \& \ 0 < a \rightarrow 0 < ab^c + ad^c - a(b + d)^c + e)$

    Выражения b, d неконстантные. Третий антецедент обрабатывается оператором "провменьшеилиравно", прочие - оператором "усмменьше". Уровень срабатывания равен 5.

16. Попытка группировки двух слагаемых.

    $\forall_{abcd}(c = a - b \ \& \ 0 \leq c \ \& \ 0 < d \rightarrow 0 < a - b + d)$

    Первый антецедент, выделенный указателем "идентификатор", предпринимает попытку разложения на множители разности $a - b$, образованной двумя слагаемыми правой части доказываемого неравенства. Проверяется, что c представляет собой произведение либо степень. Второй и третий антецеденты выделены указателем "блокпроверок". Выражение d имеет не более трех слагаемых, и ни одно из них не имеет знака "минус". Выражение b неконстантное, причем в контексте содержится линейное неравенство для некоторой его переменной. Не должна усматриваться неположительность выражения a. Введен сильный ограничитель трудоемкости. Уровень срабатывания равен 5.

17. Использование неравенства из посылок, дающего константную оценку для переменной.

    $\forall_{abcd}(a < b \ \& \ 0 < d \ \& \ 0 \leq c - bd \rightarrow 0 < c - ad)$

    $\forall_{abcd}(a < b \ \& \ 0 < d \ \& \ bd + c \leq 0 \rightarrow ad + c < 0)$

    $\forall_{abcd}(0 < b - a \ \& \ 0 < d \ \& \ 0 \leq c - bd \rightarrow 0 < c - ad)$

    $\forall_{abcd}(0 < b - a \ \& \ 0 < d \ \& \ bd + c \leq 0 \rightarrow ad + c < 0)$

    Выражения b, d константные, a - переменная. Первый антецедент берется в контексте, второй обрабатывается оператором "усмменьше", третий - оператором "провменьшеилиравно". Уровень срабатывания равен 5.

18. Возведение в квадрат неравенства с одним радикалом.

    $\forall_{abc}(0 < a \ \& \ 0 \leq b \ \& \ 0 < c \ \& \ c^2 - a^2b < 0 \rightarrow c - a\sqrt{b} < 0)$

    $\forall_{abc}(0 < a \ \& \ 0 \leq b \ \& \ 0 < c \ \& \ 0 < a^2b - c^2 \rightarrow 0 < a\sqrt{b} - c)$

Выражения a, c константные. Первые три антецедента выделены указателем "блокпроверок", четвертый - обрабатывается оператором "провменьше". Указатель "спуск" блокирует альтернативные попытки. Уровень срабатывания равен 2.

Нормализатор решения строгих неравенств "уравнменьше"

Решение неравенств часто требует разбора большого числа подслучаев. Существенное ускорение вычислений возникает при погружении этого разбора подслучаев в пакетные нормализаторы. Здесь происходит разложение на множители разности частей неравенства, разбор случаев для знаков сомножителей, решение неравенств простейших типов, усмотрение несовместных систем условий. Первоначальное обращение к нормализатору делается из приема сканирования задачи, выполняющего какой-либо шаг решения, и далее нормализатор берет на себя длинную цепочку действий, продолжающих этот шаг. Созданы два таких нормализатора - "уравнменьше" и "уравнменьшеилиравно". Их приемы, как правило, дублируют уже описанные выше приемы сканирования задачи. Поэтому при перечислении приемов нормализатора "уравнменьше" ограничимся краткими пояснениями:

1. Устранение минуса в неравенстве с нулевой частью.

   $\forall_a(-a < 0 \leftrightarrow 0 < a)$

   $\forall_a(0 < -a \leftrightarrow a < 0)$

   Уровень срабатывания равен 1.

2. Устранение дроби в неравенстве с нулевой частью.

   $\forall_{ab}(a/b < 0 \leftrightarrow ab < 0)$

   $\forall_{ab}(0 < a/b \leftrightarrow 0 < ab)$

   Уровень срабатывания равен 1.

3. Устранение степени в неравенстве с нулевой частью.

   $\forall_{ab}(0 \leq a \rightarrow 0 < a^b \leftrightarrow 0 < a)$

   $\forall_{ab}(\text{числитель}(b) - \text{even}\ \&\ b - \text{rational} \rightarrow 0 < a^b \leftrightarrow \neg(a = 0))$

   $\forall_{ab}(b - \text{rational}\ \&\ \neg(\text{знаменатель}(b) - \text{even}) \rightarrow a^b < 0 \leftrightarrow \neg(\text{числитель}(b) - \text{even})\ \&\ a < 0)$

   $\forall_{ab}(b - \text{rational}\ \&\ \neg(\text{знаменатель}(b) - \text{even}) \rightarrow 0 < a^b \leftrightarrow \neg(a = 0)\ \&\ \text{числитель}(b) - \text{even}\ \vee\ \neg(\text{числитель}(b) - \text{even})\ \&\ 0 < a)$

   $\forall_{abc}(b - \text{rational}\ \&\ \neg(\text{числитель}(b) - \text{even})\ \&\ \neg(\text{знаменатель}(b) - \text{even}) \rightarrow 0 < a^b c \leftrightarrow 0 < ac)$

   Уровень срабатывания равен 1.

4. Приведение подобных членов с неизвестными.

   $\forall_{abcde}(ab + ac + d < e \leftrightarrow a(b + c) + d < e)$

   Выражения b, c известны, a - содержит неизвестные. Указатель "дробь" разрешает перестановку частей неравенства. Уровень срабатывания равен 1.

5. Неравенство с произведением в одной части и нулем в другой.

   $\forall_{ab}(ab < 0 \leftrightarrow a < 0 \;\&\; 0 < b \;\vee\; 0 < a \;\&\; b < 0)$

   $\forall_{ab}(0 < ab \leftrightarrow a < 0 \;\&\; b < 0 \;\vee\; 0 < a \;\&\; 0 < b)$

   Каждое из неравенств правой части обрабатывается нормализатором "уравнменьше" и в простейших случаях доводится до решения. В каждой конъюнкции обращение к обработке второго неравенства сопровождается дополнительными посылками - результатом решения первого неравенства. Полученные конъюнкции решений, а также вся заменяющая дизъюнкция обрабатываются пакетным нормализатором "склейканеравенств", предпринимающим попытку упрощения дизъюнктивно - конъюнктивных комбинаций неравенств. Этот норализатор - общий для строгих и нестрогих неравенств, он вынесен в отдельный подраздел раздела базы приемов, связанного с неравенствами. Уровень срабатывания равен 2.

   Если удается определить знак одного из сомножителей, применяются следующие приемы:

   $\forall_{ab}(0 < a \rightarrow ab < 0 \leftrightarrow b < 0)$

   $\forall_{ab}(a < 0 \rightarrow ab < 0 \leftrightarrow 0 < b)$

   $\forall_{ab}(0 < a \rightarrow 0 < ab \leftrightarrow 0 < b)$

   $\forall_{ab}(a < 0 \rightarrow 0 < ab \leftrightarrow b < 0)$

   $\forall_{ab}(0 \leq a \rightarrow 0 < ab \leftrightarrow \neg(a = 0) \;\&\; 0 < b)$

   Уровень срабатывания равен 1.

6. Неравенство с неизвестной дробью в одной части.

   $\forall_{abc}(b - \text{число} \;\&\; \neg(b = 0) \rightarrow c < a/b \leftrightarrow 0 < b \;\&\; bc < a \;\vee\; b < 0 \;\&\; a < bc)$

   $\forall_{abc}(b - \text{число} \;\&\; \neg(b = 0) \rightarrow a/b < c \leftrightarrow 0 < b \;\&\; a < bc \;\vee\; b < 0 \;\&\; bc < a)$

   Либо a, либо b содержит неизвестные. Первое неравенство в каждой конъюнкции из заменяющей части обрабатывается либо нормализатором "нормменьше" (если b известно), либо нормализатором "уравнменьше" (если b не известно). Второе неравенство в каждой конъюнкции обрабатывается нормализатором "уравнменьше". Дополнительно здесь вводятся посылки, полученные при обработке первого неравенства. Если b содержит неизвестные, то в целях ускорения вычислений первое неравенство, до обработки нормализатором "уравнменьше", обрабатывается нормализатором "нормусм", который предпринимает попытку использовать проверочный оператор для усмотрения истинности либо ложности. Вся заменяющая дизъюнкция обрабатывается нормализатором "нормлог". Уровень срабатывания равен 1.

7. Решение неравенств $-x < a, a < -x$.

   $\forall_{ab}(-b < a \leftrightarrow -a < b)$

   $\forall_{ab}(a < -b \leftrightarrow b < -a)$

   Выражение a известно, b - содержит неизвестные. Уровень срабатывания равен 1.

8. Решение неравенств $x + a < b, b < x + a$.

   $\forall_{abc}(b < a + c \leftrightarrow b - c < a)$

   $\forall_{abc}(a < b - c \leftrightarrow a + c < b)$

   Первый прием выполняет замену слева направо, второй - справа налево. Выражение b известно, a - идентифицируется путем группировки всех неизвестных слагаемых. Выражения a и c невырождены. Уровень срабатывания равен 1.

9. Решение неравенств $xa < b, b < xa$.

   $\forall_{abc}(ac < b \leftrightarrow c < 0 \,\&\, b/c < a \ \vee \ 0 < c \,\&\, a < b/c \ \vee \ c = 0 \,\&\, 0 < b)$

   $\forall_{abc}(b < ac \leftrightarrow c < 0 \,\&\, a < b/c \ \vee \ 0 < c \,\&\, b/c < a \ \vee \ c = 0 \,\&\, b < 0)$

   Выражения b, c известны, a идентифицируется как произведение всех неизвестных сомножителей и невырождено. Неравенства с a в заменяющем терме обрабатываются нормализатором "уравнменьше", неравенства без a - нормализатором общей стандартизации строгих неравенств "нормменьше". Уровень срабатывания равен 1.

10. Решение неравенств с модулем.

    $\forall_{ab}(a < b \,\& \, -b < a \leftrightarrow |a| < b)$

    $\forall_{ab}(b < |a| \leftrightarrow a < -b \ \vee \ b < a)$

    Первый прием применяется справа налево, второй - слева направо. Выражение b известно, a содержит неизвестные. Каждое неравенство заменяющего терма обрабатывается нормализатором "уравнменьше". Уровень срабатывания равен 1.

11. Группировка неизвестных слагаемых в одной части.

    $\forall_{ab}(a < b \leftrightarrow 0 < b - a)$

    Выражения a, b содержат неизвестные. Правая часть обрабатывается нормализатором "нормменьше". Уровень срабатывания равен 1.

12. Решение простейшего степенного неравенства с положительным показателем степени.

    $\forall_{abc}(0 < b \,\&\, 0 \le a \to c < a^b \leftrightarrow c < 0 \ \vee \ 0 \le c \,\&\, c^{1/b} < a)$

    $\forall_{abc}(0 < b \,\&\, 0 \le a \to a^b < c \leftrightarrow 0 < c \,\&\, a < c^{1/b})$

    Выражения b, c известны, a содержит неизвестные. Неравенство с a в заменяющем терме обрабатывается нормализатором "уравнменьше", прочие неравенства - нормализатором "нормменьше". Уровень срабатывания равен 2.

    $\forall_{abc}(b - \text{rational} \,\&\, \text{числитель}(b) - \text{even} \,\&\, 0 < b \to c < a^b \leftrightarrow c < 0 \ \vee \ 0 \le c \,\&\, (a < -c^{1/b} \ \vee \ c^{1/b} < a))$

    $\forall_{abc}(b - \text{rational} \,\&\, \text{числитель}(b) - \text{even} \,\&\, 0 < b \to a^b < c \leftrightarrow 0 < c \,\&\, a < c^{1/b} \,\& \, -c^{1/b} < a)$

    $\forall_{abc}(b - \text{rational} \,\&\, \neg(\text{знаменатель}(b) - \text{even}) \,\&\, \neg(\text{числитель}(b) - \text{even}) \,\&\, 0 < b \to c < a^b \leftrightarrow c^{1/b} < a)$

    $\forall_{abc}(b - \text{rational} \,\&\, \neg(\text{знаменатель}(b) - \text{even}) \,\&\, \neg(\text{числитель}(b) - \text{even}) \,\&\, 0 < b \to a^b < c \leftrightarrow a < c^{1/b})$

Выражения b, c известны, a содержит неизвестные. В первых двух приемах содержащие a неравенства из заменяющей части обрабатываются нормализатором "уравнменьше". Прочие неравенства обрабатываются нормализатором "нормменьше". Уровень срабатывания равен 1.

13. Решение простейшего степенного неравенства с отрицательным показателем степени.

$\forall_{abc}(b < 0 \ \& \ 0 < a \to c < a^b \leftrightarrow c \le 0 \ \vee \ 0 < c \ \& \ a < c^{1/b})$

$\forall_{abc}(b < 0 \ \& \ 0 < a \to a^b < c \leftrightarrow 0 < c \ \& \ c^{1/b} < a)$

$\forall_{abc}(b - \text{rational} \ \& \ \text{числитель}(b) - \text{even} \ \& \ b < 0 \ \& \ \neg(a = 0) \to a^b < c \leftrightarrow 0 < c \ \& \ (a < -c^{1/b} \ \vee \ c^{1/b} < a))$

$\forall_{abc}(b - \text{rational} \ \& \ \text{числитель}(b) - \text{even} \ \& \ b < 0 \ \& \ \neg(a = 0) \to c < a^b \leftrightarrow c \le 0 \ \vee \ 0 < c \ \& \ a < c^{1/b} \ \& \ -c^{1/b} < a)$

$\forall_{abc}(b - \text{rational} \ \& \ \neg(\text{знаменатель}(b) - \text{even}) \ \& \ \neg(\text{числитель}(b) - \text{even}) \ \& \ b < 0 \ \& \ \neg(a = 0) \to c < a^b \leftrightarrow 0 < ac \ \& \ a < c^{1/b} \ \vee \ c \le 0 \ \& \ 0 < a)$

$\forall_{abc}(b - \text{rational} \ \& \ \neg(\text{знаменатель}(b) - \text{even}) \ \& \ \neg(\text{числитель}(b) - \text{even}) \ \& \ b < 0 \ \& \ \neg(a = 0) \to a^b < c \leftrightarrow 0 < ac \ \& \ c^{1/b} < a \ \vee \ a < 0 \ \& \ 0 \le c)$

Выражения b, c известны, a содержит неизвестные. Неравенства с выражением a в заменяющем терме обрабатываются нормализатором "уравнменьше", прочие - нормализаторами общей стандартизации неравенств. Уровень срабатывания равен 1.

14. Попытка разложения на множители разности частей неравенства.

$\forall_{abc}(c = b - a \to a < b \leftrightarrow 0 < c)$

Хотя бы одна из частей неравенства содержит неизвестные и отлична от переменной. Если одна из частей неравенства нулевая, то другая не имеет мультипликативного заголовка ("умножение", "дробь", "степень"). Оператор фильтра "фильтрмножителей" используется в данном приеме так же, как он использовался в аналогичном приеме сканирования задачи. Антецедент обращается к оператору "видумножение". Результат c не должен иметь сомножителя, имеющего вид суммы с дробным слагаемым, и сам не должен являться такой суммой. Уровень срабатывания равен 3.

15. Усмотрение противоречия из двух противоположных неравенств для неизвестной. Приемы заменяют неравенство на константу "ложь":

$\forall_{abc}(c < a \ \& \ 0 \le b - a \to \neg(b < c))$

$\forall_{abc}(c \le a \ \& \ 0 \le b - a \to \neg(b < c))$

$\forall_{abc}(a \le c \ \& \ 0 \le a - b \to \neg(c < b))$

$\forall_{abc}(a < c \ \& \ 0 \le a - b \to \neg(c < b))$

Выражения a, b известны, c - неизвестная. Первый антецедент берется в посылках, второй - обрабатывается проверочным оператором "усмменьшеилиравно". Уровень срабатывания равен 2.

16. Попытка раскрывания скобок с неизвестными слагаемыми.

$\forall_{abcdefg}(f = a(b + c)^d + e \to a(b + c)^d + e < g \leftrightarrow f < g)$

Показатель степени d - натуральная константа, не превосходящая 3 (допускается вырожденное единичное значение). Основание степени $b+c$ содержит неизвестные. Преобразуемая часть неравенства не имеет своим обобщенным слагаемым натуральную степень суммы, показатель которой больше 3. Антецедент обрабатывает левую часть неравенства нормализаторами "нормплюс", "стандплюс", "уравнплюс". Указатель "дробь" разрешает обработку как левой, так и правой части неравенства. Уровень срабатывания равен 2.

17. Квадратные неравенства.

$\forall_{abcdepqr}(d = b^2 + 4ac \;\&\; p = \sqrt{d} \;\&\; q = (-b-p)/2a \;\&\; r = (-b+p)/2a \rightarrow c < ae^2 + be \leftrightarrow a < 0 \;\&\; 0 \leq d \;\&\; (p < 0 \;\&\; q < e \;\&\; e < r \;\vee\; 0 \leq p \;\&\; r < e \;\&\; e < q) \;\vee\; 0 < a \;\&\; (d < 0 \;\vee\; 0 \leq d \;\&\; (p < 0 \;\&\; (e < r \;\vee\; q < e) \;\vee\; 0 \leq p \;\&\; (e < q \vee\; r < e))) \;\vee\; a = 0 \;\&\; c < be)$

$\forall_{abcdepqr}(d = b^2 + 4ac \;\&\; p = \sqrt{d} \;\&\; q = (-b-p)/2a \;\&\; r = (-b+p)/2a \rightarrow ae^2 + be < c \leftrightarrow 0 < a \;\&\; 0 \leq d \;\&\; (0 \leq p \;\&\; q < e \;\&\; e < r \;\vee\; p < 0 \;\&\; r < e \;\&\; e < q) \;\vee\; a < 0 \;\&\; (d < 0 \;\vee\; 0 \leq d \;\&\; (0 \leq p \;\&\; (e < r \;\vee\; q < e) \;\vee\; p < 0 \;\&\; (e < q \vee\; r < e))) \;\vee\; a = 0 \;\&\; be < c)$

Выражения a, b, c известны, e содержит неизвестные. Дискриминат d отличен от константы 0, причем не усматривается его отрицательность. Антецеденты выделены указателем "идентификатор".

$\forall_{abcde}(b^2 + 4ac = 0 \rightarrow ae^2 + be < c \leftrightarrow a < 0 \;\&\; \neg(e = -b/2a) \;\vee a = 0 \;\&\; 0 < c)$

$\forall_{abcde}(b^2 + 4ac = 0 \rightarrow c < ae^2 + be \leftrightarrow 0 < a \;\&\; \neg(e = -b/2a) \;\vee a = 0 \;\&\; c < 0)$

$\forall_{abcde}(d = b^2 + 4ac \;\&\; d < 0 \rightarrow ae^2 + be < c \leftrightarrow a < 0)$

$\forall_{abcde}(d = b^2 + 4ac \;\&\; d < 0 \rightarrow c < ae^2 + be \leftrightarrow 0 < a)$

Уровень срабатывания всех приемов равен 1.

18. Группировка слагаемых с одинаковым неизвестным радикалом.

$\forall_{abcdefghpq}(a = fg \;\&\; e = fh \rightarrow ab^{c/d} + eb^{c/d} + p < q \leftrightarrow f(g+h)b^{c/d} + p < q)$

Первые два антецедента выделены указателем "идентификатор". Они находят общий делитель f выражений a, e. Остаточные произведения g, h не должны иметь своими множителями неизвестные радикалы и степени с неизвестными показателями. Выражения c, d не содержат неизвестных, выражение b - содержит. Прием подготавливает возможность срабатывания других приемов, возводящих неравенство с неизвестными радикалами в квадрат. Указатель "дробь" разрешает перестановку частей неравенства. Уровень срабатывания равен 1.

19. Возведение в квадрат неравенства с одним либо двумя радикалами.

$\forall_{abcde}(0 \leq c \;\&\; e = a^2c - (d-b)^2 \;\&\; a \leq 0 \rightarrow a\sqrt{c} + d < b \leftrightarrow d < b \;\vee\; 0 < e)$

$\forall_{abcde}(0 \leq c \;\&\; e = a^2c - (d-b)^2 \;\&\; 0 \leq a \rightarrow b < a\sqrt{c} + d \leftrightarrow b < d \;\vee\; 0 < e)$

$\forall_{abcde}(0 \leq c \;\&\; e = a^2c - (d-b)^2 \;\&\; 0 < a \rightarrow a\sqrt{c} + d < b \leftrightarrow d < b \;\&\; e < 0)$

$\forall_{abcde}(0 \leq c \;\&\; e = a^2c - (d-b)^2 \;\&\; a < 0 \rightarrow b < a\sqrt{c} + d \leftrightarrow b < d \;\&\; e < 0)$

Выражение c содержит неизвестные, выражение d имеет не более одного слагаемого, имеющего своим сомножителем неконстантный радикал. Выражение c - самое длинное из находящихся под радикалом выражений неравенства. Уровень срабатывания равен 2.

20. Потенцирование логарифмических неравенств.

    $\forall_{abcde}(0 < b \;\&\; \neg(b-1=0) \rightarrow a\log_b c + d < e \leftrightarrow 0 < b-1 \;\&\; 0 < b^e - c^a b^d \;\vee\; b-1 < 0 \;\&\; b^e - c^a b^d < 0)$

    Хотя бы одно из выражений b, c содержит неизвестные. Либо a известно, либо неравенство имеет степень с неизвестным показателем, не расположенную внутри выражения $\log_b c$. Каждое неизвестное слагаемое выражения d либо является произведением известного коэффициента на логарифм по основанию b, либо имеет неизвестный сомножитель, являющий также сомножителем числителя слагаемого показателя некоторой степени, встречающейся в неравенстве. Указатель "дробь" разрешает перестановку частей неравенства. Уровень срабатывания равен 3.

21. Логарифмирование показательных неравенств.

    $\forall_{abcde}(0 < b \;\&\; e = d - \log_b(-c/a) \rightarrow 0 < ab^d + c \leftrightarrow 0 < a \;\&\; 0 \leq c \;\vee\; b-1 = 0 \;\&\; 0 < a+c \;\vee\; 0 < a \;\&\; c < 0 \;\&\; (0 < b-1 \;\&\; 0 < e \;\vee\; b-1 < 0 \;\&\; e < 0) \;\vee\; a < 0 \;\&\; 0 < c \;\&\; (0 < b-1 \;\&\; e < 0 \;\vee\; b-1 < 0 \;\&\; 0 < e) \;\vee\; a = 0 \;\&\; 0 < c)$

    Выражение d содержит неизвестные, b - не содержит. Выражение c не является суммой. Каждый неизвестный сомножитель выражений a, c представляет собой степень с известным основанием. Указатель "дробь" предусматривает возможность перестановки частей неравенства. Неравенства заменяющей части обрабатываются нормализаторами общей стандартизации. Второй антецедент также обращается к нормализаторам общей стандартизации. Кроме того, он предпринимает попытку разложить выражения a, c на множители. Уровень срабатывания равен 2.

    $\forall_{abc}(0 < a \rightarrow 0 < a^b - a^c \leftrightarrow 0 < 1-a \;\&\; 0 < c-b \;\vee\; 1-a < 0 \;\&\; c-b < 0)$

    Либо b, либо c содержит неизвестные. Допускается перестановка частей неравенства. Уровень срабатывания равен 1.

22. Переход к тангенсу половинного угла в тригонометрических неравенствах. Если в неравенстве встречаются только неизвестные тригонометрические операции некоторого аргумента d, а также, быть может, $d/2$, причем не усматривается, что их можно привести к аргументу $2d$, то предпринимается попытка выразить все эти операции через тангенс $d/2$:

    $\forall_{abcde}(c = a \;\&\; e = d/2 \rightarrow a < b \leftrightarrow \neg(\cos e = 0) \;\&\; c < b \;\vee\; \cos e = 0 \;\&\; a < b)$

    На этой теореме созданы два различных приема, отличающиеся выбором точки инициализации: в одном случае это вхождение синуса, в другом - косинуса. Ограничимся рассмотрением приема, связанного с синусом, так как второй прием аналогичен. Этот прием имеет указатель "контекст(позиция(x7 корень) вид(x7 синус(x4))не(контекст(подтерм(степень(теквхожд(x7)x8)) четное (x8))))", определяющий точку инициализации - вхождение выражения $\sin d$, не являющегося основанием степени с четным показателем. Проверяется, что d, a содержат неизвестные. Ввиду группировки неизвестных слагаемых в одной части неравенства, выражение b будет известным. Далее проверяется выполнение следующих требований:

    (a) Неравенство содержит вхождение $\cos d$, не являющееся основанием степени с четным показателем, либо вхождение $\operatorname{tg} e$, где e - результат преобразования $d/2$ к виду суммы.

(b) Неравенство не содержит неизвестной тригонометрической операции, отличной от операций $\sin d, \cos d, \operatorname{tg} d, \operatorname{tg} e$.

(c) Неизвестные тригонометрические операции не встречаются в качестве оснований степеней, имеющих показатели, отличные от 2 и 3.

(d) Неравенство не содержит модулей (их исключение будет предприниматься вне нормализатора, путем разбора случаев или иными средствами).

Оба антецедента выделены указателем "идентификатор". Второй обрабатывает выражение $d/2$ нормализаторами "нормдробь", "стандплюс". Это обеспечивает преобразование дроби с суммой в числителе к виду суммы дробей, как того требует стандартизация аргументов тригонометрических операций. Первый антецедент, собственно, выражает неизвестную часть неравенства a через тангенс $d/2$. Это делается с помощью нормализатора "половинныйугол", который будет рассмотрен в нижеследующих разделах, посвященных тригонометрии. Заменяющая часть рассматривает два подслучая. Если $\cos e \neq 0$, т.е. $\operatorname{tg} e$ определен, то неравенство $a < b$ заменяется на $c < b$. Если же тангенс не определен, то старое неравенство сохраняется, но сопровождается уравнением $\cos e = 0$. Неравенство $c < b$ обрабатывается нормализатором "уравнменьше", неравенство $a < b$ оставляется без изменений. Обращение к нормализатору "нормусм" для $\cos e = 0$ позволяет в очевидных ситуациях сразу исключить разбор случаев. Указатель "дробь" разрешает попытку применения приема, если a является правой частью неравенства. Уровень срабатывания равен 2.

23. Линейная комбинация синуса и косинуса в неизвестной части неравенства.

$\forall_{abc}(a\sin c + a\cos c < b \leftrightarrow \sqrt{2}a\sin(c+\pi/4) < b)$

$\forall_{abc}(a\cos c - a\sin c < b \leftrightarrow \sqrt{2}a\cos(c+\pi/4) < b)$

Выражения a, b известны, c - не известно.

$\forall_{ab}(\sqrt{3}\cos a - \sin a < b \leftrightarrow 2\cos(a+\pi/6) < b)$

$\forall_{ab}(\sqrt{3}\cos a + \sin a < b \leftrightarrow 2\cos(a-\pi/6) < b)$

$\forall_{ab}(\sqrt{3}\sin a + \cos a < b \leftrightarrow 2\sin(a+\pi/6) < b)$

$\forall_{ab}(\sqrt{3}\sin a - \cos a < b \leftrightarrow 2\sin(a-\pi/6) < b)$

Выражение b известно, a содержит неизвестные.

$\forall_{abcx}(a\sin x + a\cos x + b\sin x\cos x < c \leftrightarrow 2\sqrt{2}\sin(x+\pi/4) + 2b(\sin(x+\pi/4))^2 < b + 2c)$

$\forall_{abcx}(a\cos x - a\sin x + b\sin x\cos x < c \leftrightarrow 2\sqrt{2}\cos(x+\pi/4) + 2b(\cos(x+\pi/4))^2 < b + 2c)$

Выражения a, b, c известны, x содержит неизвестные. Во всех приемах допускается перестановка частей неравенства. Уровень срабатывания равен 1.

24. Усмотрение положительности натуральной неизвестной.

$\forall_n(n$ – натуральное & $m \leq 0 \rightarrow \neg(n < m))$

Переменная n идентифицируется с неизвестной. Первый антецедент берется в посылках, второй - обрабатывается проверочным оператором. Уровень срабатывания равен 3.

Нормализатор строгих неравенств "нормменьше"

Нормализатор предназначен для простейшей стандартизации строгих неравенств. Обычно он применяется к неравенствам без неизвестных. Используются следующие приемы:

1. Устранение минуса в неравенстве с нулевой противоположнгой частью.

   $\forall_a(0 < -a \leftrightarrow a < 0)$

   $\forall_a(-a < 0 \leftrightarrow 0 < a)$

   Уровень срабатывания равен 1.

2. Устранение множителя определенного знака в неравенстве с нулевой противоположной частью.

   $\forall_{ab}(0 < a \rightarrow 0 < ab \leftrightarrow 0 < b)$

   $\forall_{ab}(a < 0 \rightarrow 0 < ab \leftrightarrow b < 0)$

   $\forall_{ab}(0 < a \rightarrow ab < 0 \leftrightarrow b < 0)$

   $\forall_{ab}(a < 0 \rightarrow ab < 0 \leftrightarrow 0 < b)$

   Антецедент обрабатывается проверочным оператором. Введен умеренный ограничитель трудоемкости. Уровень срабатывания равен 2.

   $\forall_{ab}(a \leq 0 \rightarrow ab < 0 \leftrightarrow 0 < b \ \&\ a < 0)$

   $\forall_{ab}(0 \leq a \rightarrow ab < 0 \leftrightarrow b < 0 \ \&\ 0 < a)$

   $\forall_{ab}(a \leq 0 \rightarrow 0 < ab \leftrightarrow b < 0 \ \&\ a < 0)$

   $\forall_{ab}(0 \leq a \rightarrow 0 < ab \leftrightarrow 0 < b \ \&\ 0 < a)$

   Аналогично предыдущему, но уровень срабатывания равен 3.

3. Устранение дроби в неравенстве с нулевой частью.

   $\forall_{ab}(0 < a/b \leftrightarrow 0 < ab)$

   $\forall_{ab}(a/b < 0 \leftrightarrow ab < 0)$

   Уровень срабатывания равен 1.

4. Устранение степени в неравенстве с нулевой частью.

   $\forall_{ab}(b-\text{rational} \ \&\ \neg(\text{знаменатель}(b)-\text{even}) \rightarrow 0 < a^b \leftrightarrow \neg(a = 0) \ \&\ \text{числитель}(b)-\text{even} \vee \neg(\text{числитель}(b) - \text{even}) \ \&\ 0 < a)$

   $\forall_{ab}(b - \text{rational} \ \&\ \neg(\text{знаменатель}(b) - \text{even}) \rightarrow a^b < 0 \leftrightarrow \neg(\text{числитель}(b) - \text{even}) \ \&\ a < 0)$

   $\forall_{ab}(\text{числитель}(b) - \text{even} \ \&\ b - \text{rational} \rightarrow 0 < a^b \leftrightarrow \neg(a = 0))$

   Антецеденты обрабатываются проверочным оператором, уровень срабатывания равен 1.

   $\forall_{ab}(0 \leq a \rightarrow 0 < a^b \leftrightarrow 0 < a)$

   Антецедент обрабатывается проверочным оператором, уровень срабатывания равен 2.

5. Усмотрение истинности неравенства при помощи оператора "усмменьше".

   $\forall_{ab}(a < b \to a < b)$

   Антецедент обрабатывается оператором "усмменьше". Введен средний ограничитель трудоемкости. Уровень срабатывания равен 3.

6. Усмотрение ложности неравенства при помощи оператора "усмменьшеилиравно".

   $\forall_{ab}(a \le b \to \neg(b < a))$

   Аналогично предыдущему.

7. Возведение в квадрат частей неравенства с неизвестными радикалами.

   $\forall_{abc}(0 \le a \ \& \ 0 \le b \to 0 < (a-b)c \leftrightarrow 0 < (a^2-b^2)c)$

   $\forall_{abc}(0 \le a \ \& \ 0 \le b \to (a-b)c < 0 \leftrightarrow (a^2-b^2)c < 0)$

   Эти приемы применяются лишь в тех случаях, когда имеется комментарий "квадркорень". Антецеденты обрабатываются проверочными операторами. Выражение a не имеет заголовка "плюс". Хотя бы одно из выражений a, b имеет своим множителем квадратный радикал, содержащий неизвестные, и ни одно из них не имеет своим множителем степень с неизвестным основанием, показатель которой отличен от 1/2. Уровень срабатывания равен 1.

Нормализатор строгих неравенств для параметров "стандменьше"

Нормализатор используется при редактировании ответа задачи на описание. Он выполняет действия, ориентированные на попытку разрешить относительно параметров неравенство без неизвестных. Вообще говоря, не требуется, чтобы эти действия привели к фактическому разрешению - неравенство для параметров может оказаться сложнее неравенства для неизвестных и даже вообще не решаться элементарными средствами. Просто по исчерпании возможностей приемов выдается текущий результат. Существенной особенностью нормализатора является то, что он применяется в контекстах, где сопровождение по о.д.з. может оказаться нарушенным. Это происходит из-за того, что при редактировании ответа сопровождающие условия переформулируются в компактном виде, и непосредственное усмотрение ранее очевидных требований может оказаться недоступным. Для преодоления данной трудности в ряде приемов нормализатора вместо обращений к проверочным операторам используется усиленная версия проверки неравенства - обращение к нормализатору "стандменьше" или "стандменьшеилиравно" для получения, после цепочки эквивалентных преобразований, логической константы "истина" либо "ложь". Нормализатор имеет следующие приемы:

1. Устранение минуса в неравенстве с нулевой противоположной частью.

   $\forall_a(a < 0 \leftrightarrow 0 < -a)$

   $\forall_a(-a < 0 \leftrightarrow 0 < a)$

   В первом приеме замена выполняется справа налево. Уровни срабатывания равны 1.

2. Устранение множителя определенного знака в неравенстве с нулевой противоположной частью.

$\forall_{ab}(0 < a \to 0 < ab \leftrightarrow 0 < b)$

$\forall_{ab}(0 < a \to ab < 0 \leftrightarrow b < 0)$

Выражение a константное. Антецедент обрабатывается проверочным оператором, уровень срабатывания равен 1.

$\forall_{ab}(0 < a \to 0 < ab \leftrightarrow 0 < b)$

$\forall_{ab}(0 < a \to ab < 0 \leftrightarrow b < 0)$

Выражение a неконстантное. Антецедент обрабатывается проверочным оператором, введен достаточно сильный ограничитель трудоемкости. Уровень срабатывания равен 2.

$\forall_{ab}(a < 0 \to ab < 0 \leftrightarrow 0 < b)$

$\forall_{ab}(a < 0 \to 0 < ab \leftrightarrow b < 0)$

Аналогично предыдущему, но никаких ограничений на a не накладывается.

$\forall_{ab}(a \leq 0 \to ab < 0 \leftrightarrow 0 < b \;\&\; a < 0)$

$\forall_{ab}(0 \leq a \to ab < 0 \leftrightarrow b < 0 \;\&\; 0 < a)$

$\forall_{ab}(a \leq 0 \to 0 < ab \leftrightarrow b < 0 \;\&\; a < 0)$

$\forall_{ab}(0 \leq a \to 0 < ab \leftrightarrow 0 < b \;\&\; 0 < a)$

Выражение a неконстантное. Антецедент обрабатывается проверочным оператором. Ограничитель трудоемкости - тот же, что и выше. Уровень срабатывания равен 3.

3. Устранение дроби в неравенстве с нулевой частью. Приемы те же, что в нормализаторе "нормменьше".

4. Устранение степени в неравенстве с нулевой частью.

   $\forall_{ab}(a - \text{число} \;\&\; b - \text{rational} \;\&\; \neg(\text{знаменатель}(b) - \text{even}) \to 0 < a^b \leftrightarrow \neg(a = 0) \;\&\; \text{числитель}(b) - \text{even} \;\vee\; \neg(\text{числитель}(b) - \text{even}) \;\&\; 0 < a)$

   $\forall_{ab}(b - \text{rational} \;\&\; \neg(\text{знаменатель}(b) - \text{even}) \to a^b < 0 \leftrightarrow \neg(\text{числитель}(b) - \text{even}) \;\&\; a < 0)$

   $\forall_{ab}(\text{числитель}(b) - \text{even} \;\&\; b - \text{rational} \to 0 < a^b \leftrightarrow \neg(a = 0))$

   Уровень срабатывания равен 1.

   $\forall_{ab}(0 \leq a \to 0 < a^b \leftrightarrow 0 < a)$

   Антецедент обрабатывается проверочным оператором. Уровень срабатывания равен 2.

   $\forall_{ab}((0 \leq a) = \text{истина} \to 0 < a^b \leftrightarrow 0 < a)$

   Антецедент выделен указателем "идентификатор". Он проверяет истинность неравенства $0 \leq a$ путем обращения к нормализатору "стандменьшеилиравно". В контекстах с разрушенным "явным" сопровождением по о.д.з. такая проверка может привести к успеху, даже если проверочный оператор ее не обеспечивает. Уровень срабатывания равен 3.

5. Группировка всех ненулевых слагаемых в одной части неравенства.

   $\forall_{ab}(0 < b - a \leftrightarrow a < b)$

Замена выполняется справа налево. Выражения a, b отличны от нуля. Наличие комментария "явное" блокирует применение приема, если одна из частей неравенства представляет собой переменную, а другая - константное выражение. Разность $b - a$ обрабатывается нормализатором "видумножение", снабженным комментариями, блокирующими часть его приемов. Введен достаточно сильный ограничитель трудоемкости, несколько ослабляемый при наличии дробных слагаемых. Уровень срабатывания равен 1.

6. Попытка разложить на множители разность частей неравенства. Прием аналогичен предыдущему, но блокировка приемов оператора "видумножение" несколько видоизменена, снят ограничитель трудоемкости и уровень срабатывания поднят до 4:

   $\forall_{abc}(c = b - a \rightarrow a < b \leftrightarrow 0 < c)$

   Если одна из частей неравенства нулевая, то другая не должна иметь мультипликативный заголовок. Антецедент обращается к оператору "видумножение". Результат обращения должен иметь мультипликативный заголовок. При отсутствии комментария "явное" это требование ослабляется: допускается также случай, когда c оказывается короче хотя бы одного из выражений a, b.

7. Решение квадратного неравенства с константными коэффициентами.

   $\forall_{abcde}(d = b^2 - 4ac \rightarrow ae^2 + be + c < 0 \leftrightarrow 0 < a \;\&\; 0 < d \;\&\; 0 < 2ae + b + \sqrt{d} \;\&\; 2ae + b - \sqrt{d} < 0 \;\vee\; a < 0 \;\&\; (d < 0 \;\vee\; 0 \leq d \;\&\; (0 < 2ae + b - \sqrt{d} \;\vee\; 2ae + b + \sqrt{d} < 0)) \;\vee\; a = 0 \;\&\; be + c < 0)$

   Коэффициенты a, b, c - константные выражения, e - неконстантное. Антецедент выделен указателем "идентификатор". Не должна усматриваться отрицательность дискриминанта d. Указатель "дробь" разрешает перестановку частей неравенства. Неравенства в заменяющей части обрабатываются нормализаторами общей стандартизации. Суммы с выражением e обрабатываются нормализатором "видумножение". Уровень срабатывания равен 4.

   $\forall_{abcde}(d = b^2 - 4ac \;\&\; d < 0 \rightarrow ae^2 + be + c < 0 \leftrightarrow a < 0)$

   $\forall_{abcde}(b^2 - 4ac = 0 \rightarrow ae^2 + be + c < 0 \leftrightarrow a < 0 \;\vee\; a = 0 \;\&\; 0 < c)$

   Аналогично предыдущему.

8. Усмотрение ложности неравенства при помощи оператора "усмменьшеилиравно".

   $\forall_{ab}(a \leq b \rightarrow \neg(b < a))$

   Уровень срабатывания равен 3.

9. Усмотрение истинности неравенства при помощи оператора "усмменьше".

   $\forall_{ab}(a < b \rightarrow a < b)$

   Уровень срабатывания равен 3.

10. Возведение неравенства с радикалами в квадрат.

    $\forall_{abcde}(0 \leq c \;\&\; e = a^2 c - (d - b)^2 \rightarrow a\sqrt{c} + d < b \leftrightarrow d < b \;\&\; (e < 0 \;\vee\; a < 0) \;\vee\; 0 < e \;\&\; a < 0)$

$\forall_{abcde}((0 \leq c) =$ истина $\& \ e = a^2c - (d-b)^2 \rightarrow a\sqrt{c} + d < b \leftrightarrow d < b \ \& \ (e < 0 \ \vee \ a < 0) \ \vee \ 0 < e \ \& \ a < 0)$

Вторая версия, с усиленной проверкой неравенства $0 \leq c$ при помощи нормализатора "стандменьшеилиравно", применяется при наличии комментария "явное", первая - при его отсутствии. Выражение c должно быть неконстантным. Число слагаемых выражения d, имеющих своим множителем неконстантный радикал второй степени, не должно быть более 1. При этом $\sqrt{c}$ - самый длинный такой радикал. Указатель "дробь" разрешает перестановку частей неравенства. Уровень срабатывания равен 2.

11. Неравенство с произведением в одной части и нулем в другой.

    $\forall_{ab}(ab < 0 \leftrightarrow a < 0 \ \& \ 0 < b \ \vee 0 < a \ \& \ b < 0)$

    $\forall_{ab}(0 < ab \leftrightarrow a < 0 \ \& \ b < 0 \ \vee 0 < a \ \& \ 0 < b)$

    Уровень срабатывания равен 4.

12. Возведение в степень двучленного неравенства. Предпринимается разрешение неконстантного основания степени относительно константных выражений:

    $\forall_{abc}(b - \mathrm{rational} \ \& \ \neg(\text{знаменатель}(b) - \mathrm{even}) \ \& \ \neg(\text{числитель}(b) - \mathrm{even}) \ \& \ 0 < b \rightarrow 0 < a^b + c \leftrightarrow 0 < a + c^{1/b})$

    Выражения b, c константные, a - неконстантное. Указатель "дробь" разрешает перестановку частей неравенства.

    $\forall_{abcx}(b - \mathrm{rational} \ \& \ \text{числитель}(b) - \mathrm{even} \rightarrow 0 < ax^b + c \leftrightarrow 0 < a \ \& \ (0 < c \ \vee \ c \leq 0 \ \& \ (0 < x - (-c)^{1/b}/a^{1/b} \ \vee \ x + (-c)^{1/b}/a^{1/b} < 0)) \ \vee \ a < 0 \ \& \ 0 < c \ \& \ 0 < x + c^{1/b}/(-a)^{1/b} \ \& \ x - c^{1/b}/(-a)^{1/b} < 0 \ \vee \ a = 0 \ \& \ 0 < c)$

    $\forall_{abcx}(b - \mathrm{rational} \ \& \ \text{числитель}(b) - \mathrm{even} \rightarrow ax^b + c < 0 \leftrightarrow a < 0 \ \& \ (c < 0 \ \vee \ 0 \leq c \ \& \ (0 < x - c^{1/b}/(-a)^{1/b} \ \vee \ x + c^{1/b}/(-a)^{1/b} < 0)) \ \vee \ 0 < a \ \& \ c < 0 \ \& \ 0 < x + (-c)^{1/b}/a^{1/b} \ \& \ x - (-c)^{1/b}/a^{1/b} < 0 \ \vee \ a = 0 \ \& \ c < 0)$

    Выражения a, b, c константные, x - неконстантное. Уровни срабатывания равны 1.

13. Линейные неравенства. Предпринимается разрешение неконстантной части относительно константных фрагментов:

    $\forall_{abc}(0 < b \rightarrow ab < c \leftrightarrow a < c/b)$

    $\forall_{abc}(b < 0 \rightarrow ab < c \leftrightarrow c/b < a)$

    Выражение b идентифицируется с невырожденным произведением всех константных сомножителей, c - константное, a - неконстантное. Указатель "дробь" разрешает перестановку частей неравенства.

    $\forall_{abc}(b < a + c \leftrightarrow b - c < a)$

    $\forall_{abc}(a < b - c \leftrightarrow a + c < b)$

    Первый прием применяется слева направо, второй - справа налево. Выражение c идентифицируется с невырожденной суммой всех константных слагаемых, b - константное, a - неконстантное.

    $\forall_{ab}(a < -b \leftrightarrow b < -a)$

    $\forall_{ab}(-b < a \leftrightarrow -a < b)$

Выражение a константное, b - неконстантное. Во всех случаях преобразуемое неравенство не содержит неизвестных, имеется комментарий "явное", а уровень срабатывания равен 5.

14. Усмотрение истинности неравенства, явно разрешенного относительно параметра. При этом используется аналогичное неравенство из контекста, усиливающее данное неравенство:

    $\forall_{abc}(0 < c - b \mathbin{\&} a < b \rightarrow a < c)$

    $\forall_{abc}(0 < b - c \mathbin{\&} b < a \rightarrow c < a)$

    Выражение a идентифицируется с переменной; b, c - константные выражения. Первый антецедент обрабатывается проверочным оператором, второй - берется в контексте, причем его неравенство может быть нестрогим. Имеется комментарий "явное". Уровень срабатывания равен 3.

15. Потенцирование логарифмических неравенств.

    $\forall_{abcde}((0 < b) = \text{истина} \mathbin{\&} (b-1=0) = \text{ложь} \rightarrow a\log_b c + d < e \leftrightarrow 0 < b-1 \mathbin{\&} 0 < b^e - c^a b^d \ \vee\ b - 1 < 0 \mathbin{\&} b^e - c^a b^d < 0)$

    Хотя бы одно из выражений b, c неконстантное. Либо выражение a константное, либо неравенство имеет вхождение степени с неконстантным показателем, не расположенное внутри $\log_b c$. Каждое неконстантное слагаемое выражения d должно либо представлять собой произведение константного множителя на логарифм по основанию b, либо иметь общий неконстантный множитель со слагаемым показателя некоторой степени, входящей в неравенство. Антецеденты выделены указателями "идентификатор"; они обеспечивают проверку требований для основания логарифма в ситуации, когда явное сопровождение по о.д.з. может оказаться нарушенным. Неконстантные неравенства в заменяющей части обрабатываются нормализатором "стандменьше". Дизъюнкция и обе конъюнкции обрабатываются нормализатором "склейканеравенств". Допустима перестановка частей неравенства. Имеется комментарий "явное". Уровень срабатывания равен 4.

    $\forall_{abcdex}(a < 0 \mathbin{\&} b = c/a \mathbin{\&} e = 1/d^{1/b} \rightarrow 0 < a\log_x d + c \leftrightarrow b < 0 \mathbin{\&} (d-1 < 0 \mathbin{\&} (0 < x-1 \ \vee\ x - e < 0) \ \vee\ 0 < d-1 \mathbin{\&} (0 < x - e \ \vee\ x - 1 < 0) \ \vee\ d - 1 = 0) \ \vee\ 0 < b \mathbin{\&} (d-1 < 0 \mathbin{\&} 0 < x - 1 \mathbin{\&} x - e < 0 \ \vee\ 0 < d - 1 \mathbin{\&} 0 < x - e \mathbin{\&} x - 1 < 0))$

    $\forall_{abcdex}(0 < a \mathbin{\&} b = c/a \mathbin{\&} e = 1/d^{1/b} \rightarrow 0 < a\log_x d + c \leftrightarrow 0 < b \mathbin{\&} (0 < d-1 \mathbin{\&} (0 < x-1 \ \vee\ x - e < 0) \ \vee\ d - 1 < 0 \mathbin{\&} (0 < x - e \ \vee\ x - 1 < 0) \ \vee\ d - 1 = 0) \ \vee\ b < 0 \mathbin{\&} (0 < d - 1 \mathbin{\&} 0 < x - 1 \mathbin{\&} x - e < 0 \ \vee\ d - 1 < 0 \mathbin{\&} 0 < x - e \mathbin{\&} x - 1 < 0))$

    Выражения a, c, d константные, x - неконстантное. Первый антецедент обрабатывается проверочным оператором, второй и третий - выделены указателем "идентификатор". Неравенства с x в заменяющей части обрабатываются нормализатором "стандменьше". Имеется комментарий "явное". Допустима перестановка частей неравенства. Уровень срабатывания равен 2.

    $\forall_{ax}(0 < \log_x a \leftrightarrow 0 < a - 1 \mathbin{\&} 0 < x - 1 \ \vee\ a - 1 < 0 \mathbin{\&} x - 1 < 0)$

    Выражение a константное, x - неконстантное. Неравенства с x обрабатываются нормализатором "стандменьше". Имеется комментарий "явное". Допустима перестановка частей неравенства. Уровень срабатывания равен 2.

16. Исключение модуля.

    $\forall_{ab}(0 < b - |a| \leftrightarrow 0 < a + b \;\&\; 0 < b - a)$

    Неравенства в правой части обрабатываются нормализатором "стандменьше", конъюнкция - нормализатором "склейканеравенств". Имеется комментарий "явное". Уровень срабатывания равен 3.

17. Нахождение целой части.

    $\forall_{abn}(n - \text{целое} \;\&\; b = [a] \rightarrow a < n \leftrightarrow b + 1 \leq n)$

    $\forall_{abn}(n - \text{целое} \;\&\; b = [a] \;\&\; 0 < a - b \rightarrow n < a \leftrightarrow n \leq b)$

    Здесь n - переменная, a - константное выражение. Первый антецедент берется в контексте. Второй - выделен указателем "идентификатор"; он обрабатывает $[a]$ нормализатором "нормцелаячасть" и таким образом вычисляет эту целую часть. Третий антецедент в последнем приеме обрабатывается проверочным оператором. Уровень срабатывания равен 3.

18. Усмотрение знака произведения тригонометрических функций. Рассматривается следующий специальный случай:

    $\forall_{abc}(0 < a \;\&\; 0 < b \;\&\; 0 \leq \pi - a \;\&\; 0 \leq \pi - b \rightarrow c \sin(a - b) \sin(b/2 - a/2) < 0 \leftrightarrow \neg(a = b) \;\&\; 0 < c)$

    Антецеденты обрабатываются проверочными операторами. Допускается перестановка частей неравенства. Неравенство в заменяющей части обрабатывается нормализатором "стандменьше". Уровень срабатывания равен 3.

19. Приведение подобных членов.

    $\forall_{abcd}(0 < ab + ac + d \leftrightarrow 0 < a(b + c) + d)$

    Выражения b, c константные, a - неконстантное. Допускается перестановка частей неравенства. Должен иметься комментарий "явное". Уровень срабатывания равен 2.

20. Раскрывание скобок.

    $\forall_{abcdefg}(f = a(b + c)^d + e \rightarrow a(b + c)^d + e < g \leftrightarrow f < g)$

    Основание степени неконстантное, показатель - единица либо двойка. Выражение g константное. Антецедент, выделенный указателем "идентификатор", обращается к нормализатору "стандплюс". Допускается перестановка частей неравенства. Уровень срабатывания равен 1.

21. Константные неравенства.

    $\forall_{ab}(0 < a - b \rightarrow \neg(a < b))$

    Неравенство не содержит переменных. Антецедент обрабатывается проверочным оператором. Уровень срабатывания равен 1.

22. Решение тригонометрического неравенства на конечном промежутке.

    $\forall_{fghpq}(0 < x + p \;\&\; 0 < q - x \;\&\; (0 < x + p \;\&\; 0 < q - x \;\&\; x - \text{число} \;\&\; f(x) < g(x)) = h(x) \rightarrow f(x) < g(x) \leftrightarrow h(x))$

    Первые два антецедента берутся из посылок. В них x - переменная. Преобразуемое неравенство $f(x) < g(x)$ содержит ее под знаком тригонометрической операции и не содержит других переменных. Переменные f, g, h - функциональные. Третий антецедент выделен указателем "идентификатор". Он обращается к вспомогательной задаче на описание для разрешения неравенства $f(x) < g(x)$ на конечном промежутке $0 < x + p, 0 < q - x$ относительно неизвестной x. Из списка посылок вспомогательной задачи предварительно удаляются все утверждения контекста, содержащие x. Указатели "альтернатива" разрешают идентифицировать первые два антецедента с нестрогими неравенствами. Уровень срабатывания равен 1.

23. Решение неравенства с модулем.

    $\forall_{fghpq}((x - \text{число}\ \&\ f(x) < g(x)) = h(x) \to f(x) < g(x) \leftrightarrow h(x))$

    Идентификация инициируется указателем "контекст(входит(x23 параметры(фикс(0 1))))", выделяющим в преобразуемом неравенстве некоторую переменную x. Проверяется, что неравенство не имеет других переменных и что x встречается под модулем нелинейного относительно x выражения. Антецедент обращается к вспомогательной задаче на описание так же, как в предыдущем приеме. Фильтр "внешоператор(1 нормили)" требует, чтобы прием применялся лишь в ситуациях, когда обращение к нормализатору "стандменьше" произошло из нормализатора "нормили". Уровень срабатывания равен 1.

Синтезатор "верхнийлуч"

Синтезатор позволяет определить наименьшее значение параметра, начиная с которого будет выполняться заданное неравенство. Обращение к нему имеется в единственном приеме, используемом при решении задач на программирование численных процедур приближенного нахождения корней уравнений. Утверждение, реализуемое синтезатором, имеет вид "верхнийлуч(A y)", где A - неравенство, y - наименьшее значение параметра x, входящего в A. Сам этот параметр задается комментарием (переменная x), передаваемым синтезатору в качестве входных данных. Разумеется, утверждение "верхнийлуч(A y)" не относится к логическому языку предметного уровня, а является записью "технического" характера, позволяющей использовать возможности компилятора ГЕНОЛОГа. Фактически, отношение "верхнийлуч" связывает термы для неравенства A и нижней оценки y, а также список утверждений, образующих контекст проверки (т.е. список посылок синтезатора). Синтезатор имеет следующие приемы:

1. Выравнивание малых степеней.

   $\forall_{abcmnp}(0 < m - n\ \&\ c < 0\ \&\ \text{верхнийлуч}(0 < a + (b + c)x^m, p) \to \text{верхнийлуч}(0 < a + bx^m + cx^n, \max(p, 1)))$

   Здесь x - анализируемый параметр. Он не входит в выражения b, c, m, n. Усматривается наличие такого слагаемого выражения a, что при $x \to \infty$ степень x^m, а следовательно, и меньшая степень x^n, являются "o-малыми" от этого слагаемого. Первые два антецедента обрабатываются проверочными операторами, третий представляет собой рекурсивное обращение к синтезатору "верхнийлуч". Указатель "подстановка(фикс(0 1 2 3 2)x14 0)" разрешает показателю степени n принимать вырожденное значение 0. Результирующее выражение $\max(p, 1)$ обрабатывается вспомогательной задачей на упрощение. Уровень срабатывания равен 3.

2. Оценка тригонометрических множителей.

   $\forall_{abxm}$(верхнийлуч$(0 < a - |b|, p) \rightarrow$ верхнийлуч$(0 < a + b(\sin x)^m, p))$

   $\forall_{abxm}$(верхнийлуч$(0 < a - |b|, p) \rightarrow$ верхнийлуч$(0 < a + b|\sin x|^m, p))$

   Как и выше, x - анализируемый параметр. Выражение a имеет слагаемое, стремящееся к ∞ при $x \rightarrow \infty$. Вместо синуса может идентифицироваться косинус. Антецедент обрабатывается синтезатором "верхнийлуч". Уровень срабатывания равен 2.

3. Усмотрение результата для двучленного степенного неравенства.

   $\forall_{abxmn}(0 < a \,\&\, b < 0 \,\&\, 0 < m-n \rightarrow$ верхнийлуч$(0 < ax^m+bx^n, \max((-b/a)^{1/(m-n)}, 1)))$

   Уровень срабатывания равен 1.

4. Остаточная выдача бесконечного значения.

   $\forall_a$(верхнийлуч$(a, \infty))$

   Уровень срабатывания равен 5.

5. Перестановка частей неравенства.

   $\forall_{ab}$(верхнийлуч$(0 < -a, b) \rightarrow$ верхнийлуч$(a < 0, b))$

   При неудаче работа синтезатора обрывается. Уровень срабатывания равен 1.

6. Отбрасывание малого положительного члена в правой части.

   $\forall_{abxmp}(0 < b \,\&$ верхнийлуч$(0 < a, p) \rightarrow$ верхнийлуч$(0 < a + bx^m, \max(p, 0)))$

   $\forall_{abmp}(0 < b \,\&$ верхнийлуч$(0 < a, p) \rightarrow$ верхнийлуч$(0 < a + b, \max(p, 0)))$

   Выражение a имеет такое слагаемое, что x^m (во втором случае 1)является от него "о-малым". Уровень срабатывания равен 4.

10.11 Приемы символа "меньшеилиравно"

Этот раздел, в значительной части, повторяет предыдущий. Там, где прием для нестрогого неравенства совершенно аналогичен разобранному ранее приему для строгого неравенства, будем ограничиваться лишь его названием или названием целой группы приемов.

Стандартизация нестрогих неравенств

1. Группировка всех ненулевых слагаемых в одной части неравенства.

2. Устранение минуса в одной части неравенства, если другая часть нулевая.

3. Отрицание нестрогого неравенства.

4. Нестрогая положительность либо отрицательность дроби.

   $\forall_{ab}(0 \leq a/b \leftrightarrow 0 \leq ab)$

   $\forall_{ab}(a/b \leq 0 \leftrightarrow ab \leq 0)$

Если неравенство входит в условие задачи на описание и содержит неизвестные, прием блокируется. В этом случае применяются другие приемы, приводимые ниже. Уровень срабатывания равен 0.

5. Перенесение нуля в левую часть, если имеются слагаемые со знаком "минус".

6. Конъюнкция нестрогого неравенства и отрицания равенства.

   $\forall_{ab}(\neg(a = b)\ \&\ a \leq b \leftrightarrow a < b)$

   $\forall_{ab}(\neg(a - b = 0)\ \&\ 0 \leq b - a \leftrightarrow a - b < 0)$

   $\forall_{ab}(\neg(a - b = 0)\ \&\ b - a \leq 0 \leftrightarrow 0 < a - b)$

   Уровень срабатывания равен 2.

7. Замена нестрогого неравенства на строгое, если усматривается отрицание равенства.

   $\forall_{ab}(\neg(a - b = 0) \rightarrow b \leq a \leftrightarrow b < a)$

   $\forall_{ab}(\neg(a = b) \rightarrow a \leq b \leftrightarrow a < b)$

   Антецедент берется в контексте. Замена выполняется даже в сопровождающих по о.д.з. утверждениях. Во втором приеме предусматривается удаление условия $\neg(a = b)$, оказывающегося после замены избыточным. Уровень срабатывания второго приема равен 0, первого - 3.

   $\forall_{ab}(\neg(a = 0)\ \&\ \neg(b = 0)\ \&\ 0 \leq ab \leftrightarrow 0 < ab)$

   Прием преобразует группу условий задачи на описание. Уровень срабатывания равен 0.

8. Замена нестрогого неравенства на равенство, если усматривается обратное неравенство.

   $\forall_{ab}(0 \leq a - b \rightarrow a = b \leftrightarrow a \leq b)$

   $\forall_{ab}(b \leq a \rightarrow a = b \leftrightarrow a \leq b)$

   Замена выполняется справа налево. В первом случае антецедент обрабатывается проверочным оператором, при достаточно сильном ограничителе трудоемкости, во втором случае - берется в контексте. Вхождение неравенства - корневое. Замена выполняется даже в сопровождающих по о.д.з. утверждениях. Приемы блокируются в следующих случаях:

   (a) Неравенство расположено в условии задачи на доказательство либо на преобразование.

   (b) Неравенство расположено в условии задачи на описание, причем одна его часть представляет собой неизвестную, не входящую в другую часть (в этом случае будут работать другие приемы).

   (c) Решается задача на описание, имеющая комментарий "стандменьше".

   Если решается либо задача на доказательство, либо задача на описание, некоторое условие которой содержит равенство, выражающее неизвестную через известные параметры, а все надтермы этого равенства суть дизъюнкции и конъюнкции, то уровень срабатывания равен 1. Иначе он равен 3.

   $\forall_{ab}(0 \leq a - b \rightarrow 0 \leq b - a \leftrightarrow a = b)$

Неравенство - либо корневое, либо расположено внутри условного выражения. Антецедент берется в контексте. Уровни срабатывания равны 1 и 3.

$\forall_{ab}(a^2 + b^2 \leq 0 \leftrightarrow a = 0 \ \& \ b = 0)$

Уровень срабатывания равен 1.

$\forall_{ij}(0 \leq j - i \rightarrow j \leq i \leftrightarrow i = j)$

Решается задача на преобразование, имеющая цель "нормварианты". Такие задачи используются при вычислении определителей. Антецедент обрабатывается проверочным оператором, введен сравнительно сильный ограничитель трудоемкости. Уровень срабатывания равен 1.

$\forall_{ij}(j \in \{i, \ldots, n\} \rightarrow j \leq i \leftrightarrow i = j)$

$\forall_{abc}(0 \leq a - 3 \ \& \ 0 \leq b - 3 \ \& \ 0 \leq c - 3 \rightarrow abc - ab - ac - bc = 0 \leftrightarrow a = 3 \ \& \ b = 3 \ \& \ c = 3)$

Антецеденты обрабатываются проверочными операторами. Уровень срабатывания равен 3.

$\forall_{ab}(a \leq b \rightarrow b \leq a \leftrightarrow a = b)$

Неравенство расположено под описателем "класс". Антецедент берется в контексте. Уровень срабатывания равен 1.

9. Устранение множителя одной части неравенства, имеющего определенный знак, если другая часть неравенства нулевая.

10. Декомпозиция неравенства с произведением в одной части и нулем в другой, если известен знак суммы либо разности сомножителей. К аналогам приемов для строгих неравенств добавлены следующие приемы:

    $\forall_{abc}(a = b + \sqrt{(b-c)(b+c)} \ \& \ 0 \leq a \rightarrow 0 \leq (b+c)(b-c) \leftrightarrow 0 \leq b + c \ \& \ 0 \leq b - c)$

    $\forall_{abc}(a = b - \sqrt{(b-c)(b+c)} \ \& \ 0 \leq a \rightarrow 0 \leq (b+c)(b-c) \leftrightarrow 0 \leq b + c \ \& \ 0 \leq b - c)$

    Первый антецедент берется в контексте, второй - обрабатывается проверочным оператором. Уровень срабатывания равен 0.

11. Устранение показателя степени в сомножителе одной из частей неравенства, если другая часть нулевая.

    $\forall_{abc}(0 \leq b \ \& \ 0 < c \rightarrow 0 \leq ab^c \leftrightarrow 0 \leq ab)$

    $\forall_{abc}(c - \mathrm{rational} \ \& \ \neg(\text{числитель}(c) - \mathrm{even}) \ \& \ \neg(\text{знаменатель}(c) - \mathrm{even}) \rightarrow 0 \leq ab^c \leftrightarrow 0 \leq ab)$

    Антецеденты обрабатываются проверочными операторами. Допускается перестановка частей неравенства. Уровень срабатывания равен 1.

12. Замена неравенства с модулем на два неравенства.

13. Исключение модуля в неравенстве.

    $\forall_{ab}(0 < b - a \rightarrow b - |a| \leq 0 \leftrightarrow a + b \leq 0)$

    Антецедент берется в контексте, причем в нем допускается нестрогое неравенство. Уровень срабатывания равен 1.

14. Попытка разложения на множители ненулевой части неравенства для параметров.

15. Усмотрение равенства произведения нулю.

    $\forall_{abcd}(0 \leq a - b \ \& \ 0 \leq c - d \to 0 \leq (a-b)(d-c) \leftrightarrow (a-b)(d-c) = 0)$

    $\forall_{abcd}(0 \leq a - b \ \& \ c - d \leq 0 \to (a-b)(d-c) \leq 0 \leftrightarrow (a-b)(d-c) = 0)$

    Преобразуемое неравенство является условием задачи на описание. Первый антецедент обрабатывается проверочным оператором, второй - берется в контексте. Уровень срабатывания равен 1.

16. Усмотрение нестрогого неравенства, дублирующего строгое.

    $\forall_{ab}(a < b \to a \leq b)$

    Прием выполняет замену нестрогого неравенства на логическую константу "истина". Созданы две его версии. Первая применяется к условию задачи на описание и срабатывает на уровнях 0 и 1, вторая - применяется к посылке задачи на доказательство либо на исследование и срабатывает на уровне 1.

17. Символы бесконечности.

18. Замена нестрогого неравенства на строгое при интегрировании. Для исключения возможных особых точек на границе области интегрирования, при преобразовании подынтегрального выражения задающие область нестрогие неравенства заменяются на строгие:

    $\forall_{ab}(a \leq b \leftrightarrow a < b)$

    Неравенство является посылкой задачи на преобразование, имеющей цель "нормИнтеграл". Переменная интегрирования x определяется по комментарию (нормИнтеграл x); проверяется ее вхождение в неравенство. Уровень срабатывания равен 0.

19. Усмотрение ложности нестрогого неравенства при наличии обратного строгого.

    $\forall_{ab}(a < b \to \neg(b \leq a))$

    Антецедент берется из контекста. Уровень срабатывания равен 1.

20. Преобразование сомножителя одной из частей неравенства к виду, для которого удается усмотреть нестрогий знак. Используется равенство из контекста:

    $\forall_{abcdepq}(\neg(p = 0) \ \& \ ap + b = 0 \ \& \ e = c - bq/p \to 0 \leq (aq + c)d \leftrightarrow 0 \leq de)$

    Второй антецедент содержится в контексте. Слагаемое его левой части имеет неконстантное произведение a множителей, общих с некоторым слагаемым aq сомножителя правой части неравенства. Не усматривается ни одно из неравенств $aq + c \leq 0, 0 \leq aq + c$. Первый антецедент обрабатывается проверочным оператором, третий - выделен указателем "идентификатор". Он упрощает с помощью нормализатора "видумножение" результат подстановки в $aq + c$ выражения для a, найденного из второго антецедента. Проверяется, что усматривается одно из неравенств $e \leq 0, 0 \leq e$. Прием применяется к неравенствам, расположенным в условии задачи на доказательство либо на преобразование. Допускается перестановка частей неравенства. Уровень срабатывания равен 5.

21. Усмотрение ложности нестрогого неравенства для дроби специального вида.

    $\forall_{ab}(0 < a \ \& \ 0 < b \rightarrow \neg((a+b)/(a-b) \leq 1))$

    Антецеденты обрабатываются проверочными операторами. Уровень срабатывания равен 2.

22. Исключение дроби с приведением подобных членов.

    $\forall_{abcpqrs}(ar + bs < 0 \rightarrow c \leq (ap+bq)/(ar+bs) \leftrightarrow 0 \leq (cr-p)a + (cs-q)b)$

    $\forall_{abcpqrs}(0 < ar + bs \rightarrow c \leq (ap+bq)/(ar+bs) \leftrightarrow (cr-p)a + (cs-q)b \leq 0)$

    Выражения c, p, q, r, s идентифицируются с десятичными константами. Антецедент обрабатывается проверочным оператором. Уровень срабатывания равен 3.

23. Усмотрение в дизъюнкции двух целочисленных неравенств отрицания равенства.

    $\forall_{mnx}(x$ – целое & m – целое & n – целое & $n - m = 2 \rightarrow x \leq m \ \vee \ n \leq x \leftrightarrow \neg(x = m+1))$

    Первый антецедент берется в контексте, второй и третий - обрабатываются проверочными операторами, четвертый - выделен указателем "идентификатор". Если x - неизвестная, а неравенство находится в условии задачи на описание, то прием блокируется. Уровень срабатывания равен 3.

24. Группировка константных слагаемых в одной части неравенства.

    $\forall_{abcd}(a + b \leq c + d \leftrightarrow a - c + b \leq d)$

    Выражения a, c - константные и ненулевые, d - не константное. b может вырождаться в 0. Уровень срабатывания равен 2.

25. Устранение двух минусов.

    $\forall_{ab}(-a \leq -b \leftrightarrow b \leq a)$

    Одно из выражений a, b должно являться числовым атомом. Уровень срабатывания равен 0.

26. Исключение логарифма. Аналогично случаю строгих неравенств.

27. Замена сопровождающего нестрогого неравенства на строгое, если последнее усматривается из сопровождаемого неравенства.

    $\forall_{abc}(0 \leq b \ \& \ 0 < c \ \& \ b < a^c \rightarrow 0 \leq a \leftrightarrow 0 < a)$

    Первые два антецедента обрабатываются проверочными операторами, третий берется в контексте. Уровень срабатывания равен 1.

28. Усмотрение истинности дизъюнкции с отрицанием равенства.

    $\forall_{ab}(\neg(a = b) \ \vee \ a \leq b)$

    Уровень срабатывания равен 0.

29. Монотонность произведения.

    $\forall_{abcd}(0 \leq c \ \& \ 0 \leq d \ \& \ 0 \leq a - c \ \& \ 0 \leq b - d \rightarrow 0 \leq ab - cd)$

    Первые два антецедента обрабатываются проверочным оператором, третий и четвертый - берутся в контексте. Уровень срабатывания равен 5.

Усмотрение истинности нестрогого неравенства при помощи оператора "усмменьшеилиравно"

Аналогично случаю строгих неравенств: на теореме $\forall_{ab}(a \leq b \rightarrow a \leq b)$ основаны несколько различных приемов, обращающихся к проверочному оператору "усмменьшеилиравно" для замены нестрогого неравенства на константу "истина". В приеме, срабатывающем на уровнях 2 и 4, добавлены несколько новых фильтров. Например, блокируется применение приема к неравенствам, определяющим область варьирования связанных переменных описателя "отображение". Такие неравенства встречаются во всех термах для конечных сумм и произведений, в которых параметр меняется на заданном промежутке. Добавлены также две версии приема, срабатывающие на уровне 1. Первая анализирует неравенства в условных выражениях, вторая - при решении уравнений, извлеченных из геометрических задач.

Усмотрение ложности нестрогого неравенства при помощи оператора "усмменьше"

Прием основан на теореме $\forall_{ab}(a < b \rightarrow \neg(b \leq a))$ и аналогичен приему для строгих неравенств, срабатывающему на уровнях 3 и 4. Как и в предыдущем пункте, добавлен фильтр для внешнего описателя "отображение". Добавлены версия, срабатывающая на уровне 0 и анализирующая константное неравенство, а также версия, срабатывающая на уровне 2 в задачах на исследование, имеющих цель "контроль". Такие задачи возникают при разборе случаев и требуют усиленного контроля противоречивости посылок.

Использование оператора "провменьшеилиравно" для доказательства нестрогих неравенств

Аналогично случаю строгих неравенств.

Неравенства под кванторами

1. Исключение квантора существования. Приемы аналогичны случаю строгих неравенств:

   $\forall_{abd}(\exists_c(a < c \;\&\; c \leq b \;\&\; c - \text{число} \;\&\; d) \leftrightarrow a < b \;\&\; d)$

   $\forall_{abd}(\exists_c(a < c \;\&\; c \leq b \;\&\; \neg(c = e) \;\&\; c - \text{число} \;\&\; d) \leftrightarrow a < b \;\&\; d)$

   $\forall_{ab}(a - \text{целое} \rightarrow \exists_c(c - \text{целое} \;\&\; a \leq c \;\&\; c \leq b) \leftrightarrow a \leq b)$

   $\forall_{ab}(a - \text{целое} \rightarrow \exists_c(c - \text{целое} \;\&\; a \leq c \;\&\; c < b) \leftrightarrow a < b)$

   $\forall_a(\exists_n(n - \text{натуральное} \;\&\; a \leq n)$

   $\forall_a(\exists_n(n - \text{натуральное} \;\&\; n \leq a) \leftrightarrow 1 \leq a)$

   $\forall_a(\exists_b(b - \text{число} \;\&\; b \leq a)$

   $\forall_a(\exists_b(b - \text{число} \;\&\; a \leq b)$

   $\forall_a(\exists_n(n - \text{целое} \;\&\; n \leq a)$

   $\forall_a(\exists_n(n - \text{целое} \;\&\; a \leq n)$

   $\forall_{ab}(\exists_x(\neg(a = x) \;\&\; x - \text{число} \;\&\; b \leq x))$

   $\forall_{ab}(\exists_x(\neg(a = x) \;\&\; x - \text{число} \;\&\; x \leq b))$

2. Исключение квантора общности.

   $\forall_{abc}(\forall_x(x - \text{число} \;\&\; a \leq x \;\&\; x \leq b \rightarrow x = c) \leftrightarrow 0 < a - b \;\vee\; a = b \;\&\; a = c)$

   $\forall_{abc}(\forall_x(x - \text{число} \;\&\; a \leq x \;\&\; x < b \rightarrow x = c) \leftrightarrow 0 \leq a - b)$

   Первое неравенство может идентифицироваться как строгое.

   $\forall_{abc}(\forall_x(x - \text{число} \;\&\; a < x \;\&\; x \leq b \rightarrow x = c) \leftrightarrow 0 \leq a - b)$

   $\forall_{abc}(\forall_x(x - \text{число} \rightarrow \neg(a \leq x \;\&\; x \leq b \;\&\; c)) \leftrightarrow \neg(a \leq b) \;\vee\; \neg c)$

   Первое неравенство в заменяемом терме может идентифицироваться как строгое, и тогда неравенство в заменяющем терме - тоже строгое.

   $\forall_{abc}(\forall_x(x - \text{число} \rightarrow \neg(a < x \;\&\; x < b \;\&\; c)) \leftrightarrow \neg(a < b) \;\vee\; \neg c)$

   Первое неравенство может идентифицироваться как нестрогое.

   $\forall_{mn}(m - \text{натуральное} \;\&\; n - \text{натуральное} \rightarrow \forall_k(k - \text{натуральное} \;\&\; m \leq k \rightarrow n \leq k) \leftrightarrow n \leq m)$

   $\forall_{abc}(0 < b \rightarrow \forall_n(n - \text{натуральное} \rightarrow a \leq bn + c) \leftrightarrow a \leq b + c)$

   Приемы срабатывают на уровне 1.

   $\forall_{mAB}(m - \text{натуральное} \rightarrow \forall_k(k - \text{натуральное} \;\&\; m \leq k \rightarrow \exists_x(0 \leq k + A(x) \;\&\; B(x))) \leftrightarrow \exists_x(0 \leq m + A(x) \;\&\; B(x)))$

   Уровень срабатывания равен 2.

   $\forall_{abc}(0 < b \rightarrow \forall_n(n - \text{натуральное} \rightarrow a \leq bn + c) \leftrightarrow a \leq b + c)$

   $\forall_a(a - \text{число} \rightarrow \neg(\forall_y(y - \text{число} \rightarrow y \leq a)))$

   $\forall_a(a - \text{число} \rightarrow \neg(\forall_y(y - \text{число} \rightarrow a \leq y)))$

   Уровень срабатывания равен 0.

3. Решение неравенства под квантором.

   $\forall_{fPQ}((f(x) \leq 0) = Q(x) \rightarrow \exists_x(x - \text{число} \;\&\; P(x) \;\&\; f(x) \leq 0) \leftrightarrow \exists_x(x - \text{число} \;\&\; P(x) \;\&\; Q(x)))$

   Антецедент обращается к вспомогательной задаче на описание для разрешения неравенства относительно переменной x. Выражение $f(x)$ не равно x. Указатель "дробь(...)" предусматривает также обработку неравенств вида $0 \leq f(x)$. Уровень срабатывания равен 6.

4. Перенесение неравенства в консеквент. Аналогично случаю строгих неравенств.

5. Усмотрение кванторного равенства из двух неравенств.

   $\forall_{fgA}(\forall_x(A(x) \rightarrow f(x) \leq g(x)) \;\&\; \forall_y(A(y) \rightarrow g(y) \leq f(y)) \leftrightarrow \forall_x(A(x) \rightarrow f(x) = g(x)))$

   Прием имеет заголовок "заменатермов(второйтерм)"; он объединяет две кванторные импликации, являющиеся посылками задачи, в одну.

6. Перенесение неравенства в антецедент.

   $\forall_{ABCt}(\forall_{xy}(A(x) \;\&\; B(y) \;\&\; C(x, y) \rightarrow y \leq t(x)) \leftrightarrow \forall_{xy}(A(x) \;\&\; B(y) \;\&\; t(x) < y \rightarrow \neg C(x, y)))$

   Указатель "символ(x27)" и фильтр "равно(x27 число целое натуральное)"определяют идентификацию $B(y)$ с одним из утверждений "число(y)", "целое(y)", "натуральное(y)". Возникающие после преобразования антецеденты кванторной импликации явно разрешены относительно варьируемой переменной y. Уровень срабатывания равен 1.

Решение неравенств

Как и выше, будем ограничиваться лишь названием группы приемов, если она аналогична группе приемов для строгих неравенств:

1. Решение неравенств $-x \leq a, a \leq -x$.
2. Решение неравенств $x + a \leq b, b \leq x + a$.
3. Решение неравенств $xa \leq b, b \leq xa$.
4. Группировка в одной части всех слагаемых и последующее преобразование неравенства.
5. Нестрогая положительность либо отрицательность произведения.
6. Нестрогая положительность либо отрицательность дроби.
   $\forall_{ab}(\neg(b = 0) \to 0 \leq a/b \leftrightarrow a \leq 0 \;\&\; b < 0 \;\vee\; 0 \leq a \;\&\; 0 < b)$
   $\forall_{ab}(\neg(b = 0) \to a/b \leq 0 \leftrightarrow a \leq 0 \;\&\; 0 < b \;\vee\; 0 \leq a \;\&\; b < 0)$
   Неравенство является условием задачи на описание. Выражение a содержит неизвестные. Если задача имеет уравнение либо дизъюнктивное условие с неизвестными, прием блокируется. Уровень срабатывания равен 2.
7. Устранение дробной части неравенства с неизвестными.
8. Объединение двух нестрогих односторонних неравенств.
9. Учет невырожденности промежутка для неизвестной.
10. Усмотрение противоречия либо равенства из двух противоположных неравенств для неизвестного выражения.
    $\forall_{abc}(c \leq a \;\&\; a - b < 0 \to \neg(b \leq c))$
    $\forall_{abc}(c < a \;\&\; 0 \leq b - a \to \neg(b \leq c))$
    $\forall_{abc}(c \leq a \;\&\; 0 \leq b - a \to b \leq c \leftrightarrow c = a \;\&\; a = b)$
    Первый антецедент берется в контексте, второй - обрабатывается проверочным оператором. Имеется сильный ограничитель трудоемкости. Уровни срабатывания равны 1,3 и 6.
11. Квадратные неравенства.
12. Попытка раскрывания скобок с неизвестными слагаемыми.
13. Попытка разложения на множители разности частей неравенства.
14. Возведение в квадрат неравенств с радикалами.
15. Возведение в степень двучленных неравенств.

16. Редактирование ответа.

17. Попытка доказать либо опровергнуть неравенство с неизвестными.

18. Потенцирование логарифмических неравенств.

19. Логарифмирование показательных неравенств.

20. Существование решения неравенства. Приемы этого подраздела имеют заголовок "связка". Они используются для устранения несущественной неизвестной задачи. Все условия, в которых встречается такая неизвестная, и только они, должны идентифицироваться с утверждениями под квантором существования. Сама неизвестная идентифицируется с переменной, связываемой квантором.

    (a) Два неравенства, определяющие промежуток значений неизвестной.

    $\forall_{abc}(\exists_c(a \leq c \ \& \ c \leq b \ \& \ c - \text{число}) \leftrightarrow a \leq b)$

    $\forall_{abc}(0 < a \to \exists_x(0 < x \ \& \ ax + b \leq c \ \& \ x - \text{число}) \leftrightarrow b < c)$

    Указатель "группировка(x1)" определяет идентификацию коэффициента a второго приема путем группировки всех слагаемых с несущественной неизвестной x. Уровни срабатывания равны 0.

    (b) Неравенство для целочисленной неизвестной.

    $\forall_{ab}(a - \text{число} \ \& \ b - \text{число} \to \exists_n(n - \text{натуральное} \ \& \ a \leq n + b))$

    Указатель "вариант(...)" разрешает замену предиката "натуральное" на "целое" либо "рациональное". Уровень срабатывания равен 0.

    (c) Одно неравенство для вещественной неизвестной.

    $\forall_a(a - \text{число} \to \exists_n(n - \text{число} \ \& \ a \leq n))$

    $\forall_a(a - \text{число} \to \exists_n(n - \text{число} \ \& \ n \leq a))$

    Уровени срабатывания равны 0.

    $\forall_{ab}(a - \text{число} \ \& \ b - \text{число} \ \& \ \neg(a = 0) \to \exists_x(x - \text{число} \ \& \ 0 \leq ax + b))$

    Этот прием применяется только в задачах на описание, имеющих цель "исключ". Такая цель усиливает попытки исключения несущественных неизвестных. Уровень срабатывания равен 3.

    (d) Элементы монотонной целочисленной последовательности.

    $\forall_{af}(\text{последовательность}(f, \mathbf{N}) \ \& \ \text{возрастает}(f, \mathbf{N}) \to \exists_x(x - \text{натуральное} \ \& \ a \leq f(x)))$

    Антецеденты берутся в контексте. Уровень срабатывания равен 2.

    (e) Система односторонних неравенств для неизвестной.

    $\forall_{abn}(\exists_x(x - \text{число} \ \& \ \forall_i(i \in \{1, \ldots, n\} \to a(i) \leq x + b(i))))$

    Указатель "развертка(фикс(0 2 2))" определяет идентификацию квантора общности с системой односторонних неравенств вида $a(i) \leq x + b(i)$. Фильтры "не(входит(x23 x1))", "не(входит(x23 x2))" обеспечивают невхождение неизвестной x в выражения $a(i), b(i)$. Фильтр "меньше(0 x14)" проверяет, что число неравенств не менее 1. Уровень срабатывания равен 2.

    $\forall_{abn}(\exists_x(x - \text{натуральное} \ \& \ \forall_i(i \in \{1, \ldots, n\} \to a(i) \leq x)))$

    Аналогично предыдущему.

    $\forall_{abn}(\forall_i(i \in \{1, \ldots, n\} \to \lim_{x \to \infty}\{b(i)\} = \infty) \to \exists_x(x - \text{натуральное} \ \& \ \forall_i(i \in \{1, \ldots, n\} \to a(i) \leq b(i))))$

Указатель "развертка(фикс(0 2 2))" определяет идентификацию квантора общности $\forall_i(i \in \{1, \ldots, n\} \to a(i) \leq b(i))$ с группой нестрогих неравенств, являющихся условиями задачи на описание. Указатель "фильтр(фикс(0 2 2)и(входит(x23 x2)не(входит(x23 x1))))" задает дополнительное ограничение на отбираемые нестрогие неравенства: x должно входить в $b(i)$ и не входить в $a(i)$. Антецедент выделен указателями "идентификатор(1)" и "развертка(фикс(1))". Это означает, что при $i = 1, \ldots, n$ будут вычисляться с помощью нормализатора "нормпредел" пределы $\lim_{x\to\infty}\{b(i)\}$, которые будут сравниваться с ∞. Уровень срабатывания равен 2.

21. Сведение однородного показательного неравенства к квадратному.

22. Попытка декомпозиции неравенства с помощью операторов "верхняяоценка", "нижняяоценка".

$\forall_{abcde}(b \leq d \ \&\ c \leq e \ \&\ 0 = d + e - a \to a \leq b + c \leftrightarrow b = d \ \&\ c = e)$

Неравенство $a \leq b + c$ является условием задачи на описание. Выражения b, c содержат неизвестные, выражение a - константное. Указатели "значения(1)", "значения(2)" определяют обработку первых двух антецедентов синтезатором "верхняяоценка". Он находит константные верхние оценки d, e для выражений b, c. Третий антецедент, выделенный указателем "идентификатор", проверяет, что сумма этих оценок равна a. Уровень срабатывания равен 7.

$\forall_{abcde}(d \leq b \ \&\ e \leq c \ \&\ 0 = d + e - a \to b + c \leq a \leftrightarrow b = d \ \&\ c = e)$

Аналогично предыдущему, но используется синтезатор "нижняяоценка".

$\forall_{abcde}(0 \leq c \ \&\ 0 < a \ \&\ c \leq d \ \&\ b \leq e \ \&\ de - a = 0 \to a \leq bc \leftrightarrow b = e \ \&\ c = d)$

$\forall_{abcde}(0 \leq c \ \&\ a < 0 \ \&\ c \leq e \ \&\ d \leq b \ \&\ de - a = 0 \to bc \leq a \leftrightarrow b = d \ \&\ c = e)$

Первые два антецедента обрабатываются проверочными операторами, третий и четвертый - синтезаторами "верхняяоценка" либо "нижняяоценка", пятый - выделен указателем "идентификатор". Уровень срабатывания равен 7.

23. Попытка опровергнуть неравенство с помощью оператора "верхняяоценка".

$\forall_{abc}(b - a \leq c \ \&\ c < 0 \to \neg(a \leq b))$

Неравенство $a \leq b$ является условием задачи на описание и содержит неизвестные. Первый антецедент, используя синтезатор "верхняяоценка", находит константную верхнюю оценку c для разности правой и левой частей. Второй антецедент с помощью проверочного оператора устанавливает, что c меньше 0. Уровень срабатывания равен 7.

24. Попытка усмотреть истинность неравенства с помощью оператора "нижняяоценка".

$\forall_{abc}(c \leq b - a \ \&\ 0 \leq c \to a \leq b)$

Неравенство является условием задачи на описание и содержит неизвестное выражение вида "целаячасть(...)". Первый антецедент обрабатывается синтезатором "нижняяоценка", второй - проверочным оператором. Прием позволяет усматривать избыточность неравенства после сужения промежутка изменения неизвестной. Введен средний ограничитель трудоемкости. Уровень срабатывания равен 4.

25. Исключение внешнего квантора путем перехода к точным верхним и нижним граням.

    $\forall_{abfgh}(b = \sup(\text{образ}(\lambda_x(g(x), x - \text{число}), \text{set}_x(f(x)))) \to \forall_x(f(x) \to h(x)\ \&\ g(x) \leq a) \leftrightarrow \forall_x(f(x) \to h(x))\ \&\ b \leq a)$

    Кванторная импликация $\forall_x(f(x) \to h(x)\ \&\ g(x) \leq a)$ является условием задачи на описание, содержащим неизвестные. Термы $f(x), g(x)$ не содержат неизвестных, а выражение a - содержит. Задача не имеет цели "пример". Антецедент выделен указателем "идентификатор". Используя вспомогательную задачу на упрощение, он вычисляет точную верхнюю грань значений выражения $g(x)$ на множестве элементов x, удовлетворяющих условию $f(x)$. Проверяется, что результат упрощения не имеет заголовка sup. Допускается вырождение дополнительного условия $h(x)$. Уровень срабатывания равен 2.

    $\forall_{abfgh}(b = \inf(\text{образ}(\lambda_x(g(x), x - \text{число}), \text{set}_x(f(x)))) \to \forall_x(f(x) \to h(x)\ \&\ a \leq g(x)) \leftrightarrow \forall_x(f(x) \to h(x))\ \&\ a \leq b)$

    Аналогично предыдущему.

26. Явное разрешение неравенства под квантором относительно неизвестной.

    $\forall_{afgh}(a = (g(x) \leq h(x)) \to \forall_x(f(x)\ \&\ x - \text{число} \to g(x) \leq h(x)) \leftrightarrow \forall_x(f(x)\ \&\ x - \text{число} \to a))$

    Кванторная импликация является условием задачи на описание. Утверждение $f(x)$ не содержит неизвестных. Число неизвестных неравенства $g(x) \leq h(x)$ равно 1, причем хотя бы одна из неизвестных частей этого неравенства не является неизвестной. Указатель "контекст(неизвестная(x3)входит(x3 фикс(0 1 6)))" идентифицирует x3 с неизвестной неравенства. Антецедент выделен указателем "идентификатор". Он разрешает неравенство относительно x3 посредством вспомогательной задачи на описание. Указатель "альтернатива" допускает рассмотрение строгих неравенств. Уровень срабатывания равен 4.

27. Условие на конечность множества решений нестрогого неравенства.

    $\forall_{abnfgh}(a = \lambda_x(g(x) - f(x), h(x))\ \&\ b = \text{set}_x(h(x))\ \&\ \text{непрерывно}(a, b)\ \&\ \text{регулярно}(b)\ \&\ n{-}\text{число}\ \&\ 0 < n \to \text{card}(\text{set}_x(f(x) \leq g(x)\ \&\ h(x))) = n \leftrightarrow \exists_{uv}(\text{Max}(a, b, u, v)\ \&\ v = 0\ \&\ \text{card}u = n))$

    Утверждение $\text{card}(\text{set}_x(f(x) \leq g(x)\ \&\ h(x))) = n$ является условием задачи на описание. Здесь $h(x)$ не содержит неизвестных, неравенство - содержит. Переменная x может идентифицироваться с группой переменных. Первые два антецедента выделены указателем "идентификатор". Они определяют вспомогательную функцию a для разности частей неравенства и множество b, на котором данная функция рассматривается. Остальные антецеденты обрабатываются проверочными операторами. Устанавливается, что функция a непрерывна на множестве b и что само множество b - "регулярно", т.е. представимо в виде объединения прямых произведений непустых и неодноточечных промежутков. Утверждение $\text{Max}(a, b, u, v)$ в заменяющем терме разрешается с помощью вспомогательной задачи на описание относительно неизвестных u, v, и замена выполняется только при успехе этой попытки. Уровень срабатывания равен 1.

28. Использование кванторного неравенства для получения оценки в задаче с целью "пример". Приводимые ниже приемы имеют заголовок "подборзначений".

$\forall_{abctfA}(A(t) \mathbin{\&} \forall_x(A(x) \to 0 \leq a + f(x)) \mathbin{\&} 0 \leq b - ac \mathbin{\&} 0 < c \to 0 \leq b + cf(t))$

Неравенство $0 \leq b + cf(t)$ является условием задачи на описание, имеющей цель "пример". В контексте этого условия имеется кванторная импликация $\forall_x(A(x) \to 0 \leq a + f(x))$. Переменная f - функциональная, т.е. $f(x)$ может идентифицироваться с выражением произвольного вида. Указатель "контекст(позиция(x7 фикс(2 5 2))позиция(x9 корень)вид(x7 значение(x8 x23))вид(x9 значение(x8 x23)))" определяет идентификацию в кванторной импликации выражения $h(x)$, а в рассматриваемом неравенстве - выражения $h(t)$. Здесь переменная h - не функциональная, т.е. обычная переменная, обозначающая в задаче некоторую функцию. Таким образом, оказывается идентифицировано t. Далее выражение $f(x)$ идентифицируется как сумма всех содержащих переменную x слагаемых правой части неравенства из консеквента импликации. Указатель "группировки(x6 фикс(0 2 2))" определяет идентификацию $cf(t)$ путем группировки в правой части неравенства всех членов, делящихся на уже идентифицированное выражение $f(t)$. Первый антецедент проверяет выполнение условия $A(t)$ с помощью вспомогательной задачи на доказательство. Четвертый антецедент обрабатывается проверочным оператором. Наконец, третий антецедент, выделенный указателем "подборзначений", определяет заменяющее неравенство, используемое при попытке оценить снизу $f(t)$ выражением $-a$. Уровень срабатывания равен 5.

$\forall_{abcftmnpqy}(\forall_x(A(x) \to b(x) \leq |a(x) + c(x)f(t(x))|) \mathbin{\&} \neg(c(x) = 0) \mathbin{\&} \neg(q = 0) \mathbin{\&} \exists_x(A(x) \mathbin{\&} y = t(x) \mathbin{\&} m|c(x)| \leq n|c(x)| + b(x)|q| - |a(x)q - pc(x)|) \to m \leq n + |p + qf(y)|)$

Неравенство является условием задачи на описание, имеющей цель "пример". Переменная y идентифицируется с неизвестной. Переменная f - обычная; она обозначает некоторую рассматриваемую в задаче функцию. Переменные a, b, c, t, A функциональные. Первый антецедент берется в контексте. Он представляет собой кванторную импликацию, консеквентом которой служит неравенство для модуля, аналогичное рассматриваемому. Второй и третий антецеденты обрабатываются проверочными операторами, четвертый - выделен указателем "подборзначений". Он дает параметрическое описание, замещающее исходное неравенство при попытке вывести последнее из кванторной импликации. Указатель "вариант" разрешает использование кванторной импликации со строгим неравенством. Уровень срабатывания равен 6.

$\forall_{abcftmnpqy}(\forall_{xy}(A(x, y) \to |a(x) + b(x)f(y)| \leq c(x)) \mathbin{\&} \neg(b(x) = 0) \mathbin{\&} \neg(q = 0) \mathbin{\&} \exists_x(A(x, t) \mathbin{\&} m|b(x)| \leq n|b(x)| - |pb(x) - qa(x)| - |q|c(x)) \to m \leq n - |p + qf(t)|)$

Аналогично предыдущему, но с неизвестной идентифицируется переменная t. Переменная f - обычная, переменные a, b, c, A - функциональные.

29. Использование синтезаторов "верхняяоценка", "нижняяоценка" для устранения зависимости неравенства от заданных переменных. Аналогично случаю строгих неравенств.

30. Разное. В заключение приведем несколько приемов, используемых в редко встречающихся ситуациях.

    $\forall_{abc}(0 \leq a + c \mathbin{\&} 0 \leq c - a \to 0 \leq a \cos b + c)$

Неравенство является условием задачи на описание и содержит неизвестные. Прием заменяет его на константу "истина". Антецеденты обрабатываются проверочными операторами. Уровень срабатывания равен 0.

$\forall_{abc}(0 < c \to 0 \leq a + b/c \leftrightarrow 0 \leq ac + b)$

Неравенство является условием задачи на описание, имеющей цель "стандравно". Такие задачи используются для решения систем уравнений, выведенных в задаче на исследование, имеющей цель "известно" (например, в планиметрической задаче на вычисление). Знаменатель c идентифицируется с константой. Уровень срабатывания равен 1.

Следующие два приема преобразуют нестрогое неравенство, определяющее область интегрирования, в строгое:

$\forall_{ax}(a \leq x \leftrightarrow a < x)$

$\forall_{ax}(x \leq a \leftrightarrow x < a)$

Неравенство входит в условие задачи на описание, имеющей цель "внутренность", причем x - неизвестная, не входящая в a. Уровень срабатывания равен 0. Наконец, упомянем прием, усматривающий ответ в задаче на описание, имеющей цель "неравенства":

$x - \text{число} \ \& \ a \leq x \ \& \ x \leq b \ \& \ f(x) \leq y \ \& \ y \leq g(x) \ \& \ y - \text{число}$

Заголовок приема - "ответзадачи". Указатель "смответ(1)" выбирает в качестве точки привязки первый член конъюнкции. Все члены этой конъюнкции идентифицируются с условиями задачи на описание, причем остальные условия задачи не содержат неизвестных. Цель "неравенства" означает, что задача связана с определением области интегрирования. Переменные x, y суть различные неизвестные, выражения a, b известны, выражения $f(x), g(x)$ не содержат y. Переменные f, g функциональные. Указатели "вариант" допускают использование строгих неравенств вместо любых из имеющихся в теореме приема нестрогих. Уровень срабатывания равен 0.

Доказательство неравенств

Пункты, аналогичные случаю строгих неравенств, приводим ниже без комментариев.

1. Раскрывание скобок.
2. Сложение дробей.
3. Возведение в квадрат неравенства с одним радикалом.
4. Возведение в квадрат неравенства с двумя радикалами.
5. Разность квадратов.

   $\forall_{abcd}(0 < c \ \& \ 0 < d \ \& \ 0 \leq b \ \& \ 0 \leq a \to 0 \leq cb^2 - da^2 \leftrightarrow 0 \leq \sqrt{c}b - \sqrt{d}a)$

   Неравенство входит в условие задачи на доказательство. Выражения c, d идентифицируются как произведения всех сомножителей, являющихся десятичными числами. Антецеденты обрабатываются проверочными операторами. Допускается перестановка частей неравенства. Уровень срабатывания равен 1.

6. Разложение на множители.

7. Разбор случаев по знаку выражения под модулем.

   $\forall_a(a \leq 0 \ \vee \ 0 < a)$

   Указатель "контрольвывода(модуль(x1))" инициирует применение приема при усмотрении выражения $|a|$. Это выражение должно входить в неравенство - условие задачи на доказательство. Должна существовать переменная, входящая в неравенство вне данного вхождения модуля. Если выражений под модулем несколько, выбирается самое короткое. Выводимая приемом дизъюнкция сопровождается комментарием "разборслучаев". При наличии в неравенстве конечной суммы либо конечного произведения уровень срабатывания равен 7, иначе - равен 4.

8. Вывод следствий в посылках.

   (a) Неотрицательность разности квадратов.

   (b) Сложение неравенств.

   (c) Вывод из неравенства для суммы квадратов.

   (d) Вывод из равенства для суммы квадратов.
   $\forall_{abc}(a^2 + b^2 - c^2 = 0 \ \& \ 0 \leq a + b \rightarrow 0 \leq a + b - \sqrt{c})$
   $\forall_{abc}(a^2 + b^2 - c = 0 \rightarrow 0 \leq \sqrt{2c} - a - b)$
   $\forall_{abc}(a^2 + b^2 = c \ \& \ 0 \leq a + b \rightarrow 0 \leq a + b - \sqrt{c})$
   $\forall_{abc}(a^2 + b^2 = c \rightarrow 0 \leq \sqrt{2c} - a - b)$

   Равенство берется в посылках задачи на доказательство. Условие этой задачи содержит сумму, имеющую такие слагаемые, что a, b суть их натуральные степени (включая случай показателя единица). Выражение c константное.

   (e) Регистрация неравенства, полученного группировкой в одной части всех ненулевых членов.

   (f) Линейная комбинация неравенств для исключения переменной.

9. Неотрицательность произведения.

   $\forall_{ab}(0 \leq a \ \& \ 0 \leq b \rightarrow 0 \leq ab)$

   Неравенство входит в условие задачи на доказательство. Первый антецедент обрабатывается проверочным оператором "усмменьшеилиравно", второй - усиленным проверочным оператором "провменьшеилиравно". Эти проверки существенно ограничены по трудоемкости, причем для каждого сомножителя a отсчет трудоемкости начинается заново. В случае успеха неравенство заменяется на константу "истина". Уровень срабатывания равен 2.

   $\forall_{abcd}(0 \leq c \ \& \ 0 \leq d \ \& \ 0 \leq a - c \ \& \ 0 \leq b - d \rightarrow 0 \leq ab - cd)$

   Первые два антецедента обрабатываются проверочным оператором, третий и четвертый - берутся в контексте. Уровень срабатывания равен 5.

10. Усмотрение квадратного трехчлена. Аналогично случаю строгих неравенств, но добавлена еще одна версия приема:

    $\forall_{abcx}(0 \leq c \ \& \ b^2 - 4ac \leq 0 \rightarrow 0 \leq ax^2 + bx + c)$

    $\forall_{abcx}(0 < a \ \& \ b^2 - 4ac \leq 0 \rightarrow 0 \leq ax^2 + bx + c)$

11. Усмотрение знака кубического четырехчлена при наличии оценки для переменной.

    $\forall_{abcdex}(a < 0 \,\&\, x \leq e \,\&\, 0 \leq ae^3 + be^2 + ce + d \,\&\, 3ae^2 + 2be + c < 0 \,\&\, 0 < 6ae + 2b \to 0 \leq ax^3 + bx^2 + cx + d)$

    Неравенство является условием задачи на доказательство. Переменные a, b, c, d идентифицируются с десятичными константами. Второй антецедент обращается к синтезатору "верхняяоценка" для определения верхней численной оценки e. Остальные антецеденты выделены указателем "программа" и проверяются путем непосредственного вычисления. Они означают следующее. В точке e рассматриваемое неравенство выполнено. Первая производная от кубического четырехчлена представляет собой квадратный трехчлен, графиком которого является перевернутая парабола. В точке e эта парабола находится ниже оси абсцисс, причем положительность второй производной означает, что точка e расположена левее вершины параболы. Следовательно, на луче $(-\infty, e)$ правая часть неравенства убывала и была неотрицательна. Указатели "подстановка" предусматривают возможность обращения в 0 коэффициентов b, c. Уровень срабатывания равен 5.

12. Использование индуктивного предположения.

    $\forall_{abcd}(b + d \leq 0 \,\&\, 0 \leq a \,\&\, c - ad \leq 0 \to ab + c \leq 0)$

    Неравенство является условием задачи на доказательство. Первый антецедент представляет собой посылку этой задачи, снабженную комментарием (целое e). Такой комментарий выделяет индуктивное предположение при доказательстве индукцией по параметру e. Прием пытается использовать это индуктивное предположение. Второй антецедент обрабатывается вспомогательной задачей на доказательство, решаемой до максимального уровня 4, третий - вспомогательной задачей на доказательство, максимальный уровень которой равен максимальному уровню текущей задачи. Его левая часть предварительно обрабатывается нормализаторами раскрывания скобок и разложения на множители, причем проверяется, что результат короче левой части доказываемого неравенства. Уровень срабатывания равен 2.

    $\forall_{abcd}(0 \leq b + d \,\&\, 0 \leq a \,\&\, 0 \leq c - ad \to 0 \leq ab + c)$

    Аналогично предыдущему.

13. Использование неравенства Бернулли. Аналогично случаю строгих неравенств.

14. Использование степенной нижней оценки для части множителей факториала.

15. Использование оператора "верхняяоценка".

16. Разное.

    $\forall_{abcd}(0 \leq \sqrt{a - b}\sqrt{a + b} + c \,\&\, 0 \leq d - \sqrt{2}a \to 0 \leq b + c + d)$

    Неравенство является условием задачи на доказательство. Выражение c неконстантное. Существует параметр этого выражения, не входящий в b, d. Первый антецедент берется в посылках, второй - обрабатывается вспомогательной задачей на доказательство. Введен сильный ограничитель трудоемкости. Уровень срабатывания равен 4.

    $\forall_{abcd}(0 \leq a|\sin b| + a|\sin c| + d \,\&\, a \leq 0 \to 0 \leq a|\sin(b + c)| + d)$

$\forall_{abcd}(0 \leq a|\cos b| + a|\cos c| + d \ \& \ a \leq 0 \to 0 \leq a|\sin(b+c)| + d)$

$\forall_{abcd}(0 \leq a|\sin b| + a|\cos c| + d \ \& \ a \leq 0 \to 0 \leq a|\cos(b+c)| + d)$

$\forall_{abcd}(0 \leq a|\cos b| + a|\sin c| + d \ \& \ a \leq 0 \to 0 \leq a|\cos(b+c)| + d)$

Неравенство входит в условие задачи на доказательство, содержащее также указанные в антецеденте тригонометрические операции от b и c. Первый антецедент обрабатывается вспомогательной задачей на доказательство, второй - проверочным оператором. Указатель "альтернатива" допускает случай строгого неравенства. Уровень срабатывания равен 3.

Неравенства с целочисленной неизвестной

1. Вычисление целой части.

   $\forall_{abn}(n - \text{целое} \ \& \ b = [a] \ \& \ 0 < a - b \to a \leq n \leftrightarrow b + 1 \leq n)$

   Прежде всего, проверяется наличие в контексте утверждений с заголовками "целое" либо "натуральное". После этого с помощью проверочного оператора устанавливается, что n - целое. Проверяется также, что не очевидна целочисленность значения a. Тогда второй антецедент, выделенный указателем "идентификатор", с помощью нормализатора "нормцелаячасть" находит целую часть b выражения a. Проверяется, что выражение b не содержит символа "целаячасть". Третий антецедент, обрабатываемый проверочным оператором, проверяет, что a строго больше своей целой части. Уровень срабатывания приема равен 1.

   $\forall_{abn}(n - \text{целое} \ \& \ b = [a] \to n \leq a \leftrightarrow n \leq b)$

   Аналогично предыдущему.

   $\forall_{abn}(n - \text{целое} \ \& \ 0 < a \to 0 \leq b - an \leftrightarrow n \leq [b/a])$

   $\forall_{abn}(n - \text{целое} \ \& \ 0 < a \to 0 \leq b + an \leftrightarrow -[b/a] \leq n)$

   Неравенство расположено в посылке задачи. Выражения a, b суть десятичные константы, n - переменная. Задача имеет посылку с заголовком "целое" либо "натуральное". Первый антецедент берется в контексте (допускается также заголовок "натуральное"), второй - обрабатывается проверочным оператором. Нормализатор "нормцелаячасть" предпринимает попытку вычисления целой части. Уровень срабатывания равен 2.

2. Переход от пары неравенств к перечислению конечного перечня значений.

   $\forall_{akmn}(k - \text{целое} \ \& \ m - \text{целое} \ \& \ a = k - m + 1 \ \& \ 0 < a \to m \leq n \ \& \ n \leq k \ \& \ n - \text{целое} \leftrightarrow \exists_b(b \in \{0, \ldots, a-1\} \ \& \ n = m + b))$

   $\forall_{akmn}(k - \text{целое} \ \& \ m - \text{целое} \ \& \ a = k - m + 1 \ \& \ 0 < a \ \& \ n - \text{целое} \to m \leq n \ \& \ n \leq k \leftrightarrow \exists_b(b \in \{0, \ldots, a-1\} \ \& \ n = m + b))$

   Приемы имеют заголовок "заменаусловия(второйтерм)". Они заменяют группу условий задачи на описание (первый - условия $m \leq n, n \leq k, n -$ целое, второй - $m \leq n, n \leq k$). Выражения k, m не содержат неизвестных, n - переменная. Третий антецедент вычисляет, используя нормализаторы "нормплюс" и "нормминус", увеличенную на 1 разность между пределами для n. Результат оказывается положительной десятичной константой a. Второй прием отличается от первого лишь тем, что условие целочисленности n устанавливается проверочным оператором. Указатель "или(фикс(0 2)фикс(0 2 2 1))" определяет построение заменяющего терма в виде дизъюнкции $n = m \vee n = m{+}1 \vee \ldots \vee n = m{+}a{-}1$. Если редактируется ответ задачи, то во втором приеме переменная n должна быть несущественной неизвестной. Величина a должна быть меньше 15. Уровни срабатывания равны 1, 2 и 6. Если a больше 3, то уровень равен 6. Создана также версия первого приема для натуральных значений n.

$\forall_{abcn}(c = a + b + 1 \ \& \ 0 < c \ \& \ n - \text{целое} \ \& \ 0 \leq n + a \ \& \ 0 \leq b - n \rightarrow \exists_m(m \in \{0, \ldots, c-1\} \ \& \ n = m - a))$

Прием имеет заголовок "выводусловия" и используется для инциирования разбора случаев по значениям известного целочисленного параметра n. Четвертый и пятый антецеденты идентифицируются с условиями задачи на описание. a, b - целочисленные константы, n - переменная, не являющаяся неизвестной. Величина c, равная количеству допустимых значений n, не превосходит 5. Не имеет места этап редактирования ответа. Указатель "или(фикс(0)фикс(0 2 1))" определяет развертку квантора существования в дизъюнкцию. Эта дизъюнкция сопровождается комментарием "разборслучаев". Уровень срабатывания равен 4.

$\forall_{akmn}(k - \text{целое} \ \& \ m - \text{целое} \ \& \ a = k - m + 1 \ \& \ 0 < a \ \& \ m \leq n \ \& \ n \leq k \ \& \ n - \text{целое} \rightarrow \exists_b(b \in \{0, \ldots, a-1\} \ \& \ n = m + b))$

Прием имеет заголовок "вывод". Он инициирует разбор случаев по значениям целочисленного параметра n задачи на исследование. Пятый и шестой антецеденты идентифицируются с посылками задачи. Выражения k, m не содержат неизвестных, n - переменная. Третий антецедент, выделенный указателем "идентификатор", определяет число a допустимых значений параметра n. Оно должно быть не более 9. Квантор существования разворачивается в дизъюнкцию, снабжаемую комментарием "разборслучаев". Уровень срабатывания равен 6.

$\forall_{mnk}(k - \text{целое} \ \& \ m - \text{целое} \ \& \ n - \text{целое} \ \& \ m - k = 1 \rightarrow k/2 \leq n \ \& \ n \leq m/2 \leftrightarrow n = -[-k/2])$

$\forall_{mnk}(k - \text{целое} \ \& \ m - \text{целое} \ \& \ n - \text{целое} \ \& \ m - k = 1 \rightarrow k/2 \leq n \ \& \ n < m/2 \leftrightarrow 2|k \ \& \ n = k/2)$

Приемы имеют заголовок "заменаусловия(второйтерм)". Они заменяют на равенство два противоположных неравенства - условия задачи на описание; во втором случае равенство дополняется утверждением о четности k. Переменная n является неизвестной, выражения k, m неизвестных не содержат. Второй прием имеет две версии - точка привязки берется либо по нестрогому неравенству, либо по строгому. Уровень срабатывания равен 0.

3. Усмотрение единицы.

   $\forall_n(n - \text{натуральное} \rightarrow n \leq 1 \leftrightarrow n = 1)$

   $\forall_n(n - \text{натуральное} \rightarrow 0 \leq 1 - n \leftrightarrow n = 1)$

   Уровень срабатывания равен 0.

4. Переход от параметрического описания с конечной областью значений целочисленного параметра к дизъюнкции.

   $\forall_{fmkpx}(p = k - m \ \& \ k - \text{целое} \ \& \ m - \text{целое} \rightarrow \exists_n(n - \text{целое} \ \& \ x = f(n) \ \& \ m \leq n \ \& \ n \leq k) \leftrightarrow \exists_l(l \in \{0, \ldots, p\} \ \& \ x = f(m + l)))$

Первый антецедент, выделенный указателем "идентификатор", определяет разность p между верхним и нижним пределами изменения параметра n. Проверяется, что выражение p является десятичной константой, не превосходящей 8. Заменяющее утверждение, выделенное указателем "или", разворачивается в дизъюнкцию. Уровень срабатывания равен 1.

$\forall_{abcdfprs}(b \le c \ \& \ d \le a \ \& \ r = -[-d] \ \& \ s = [c] \ \& \ p = s - r \to \exists_n(f(n) \ \& \ n -$ целое $\& \ a \le n \ \& \ n \le b) \leftrightarrow \exists_n(f(n) \ \& \ n -$ целое $\& \ a \le n \ \& \ n \le b \ \& \ \exists_m(m \in \{0, \dots, p\} \ \& \ n = r + m)))$

Здесь первый и второй антецеденты, выделенные указателями "значение", обращаются к синтезаторам "верхняяоценка" и "нижняяоценка" для получения численных верхней и нижней оценок c, d пределов b, a изменения параметра n. Третий и четвертый антецеденты, выделенные указателем "идентификатор", определяют целочисленные константы s, r для диапазона изменения параметра. Внутренний квантор существования в заменяющем утверждении выделен указателем "или"; он разворачивается в дизъюнкцию. Чтобы прием не опережал другие преобразования общей стандартизации, его применение блокируется, если $f(n)$ имеет своим конъюнктивным членом дизъюнкцию либо равенство вида $n = t$. Уровень срабатывания равен 1.

$\forall_{abcdfprs}(b \le c \ \& \ d \le a \ \& \ r = -[-d] \ \& \ s = -[-c] - 1 \ \& \ p = s - r \to \exists_n(f(n) \ \& \ n -$ целое $\& \ a \le n \ \& \ n < b) \leftrightarrow \exists_n(f(n) \ \& \ n -$ целое $\& \ a \le n \ \& \ n < b \ \& \ \exists_m(m \in \{0, \dots, p\} \ \& \ n = r + m)))$

$\forall_{abcdfprs}(b \le c \ \& \ d \le a \ \& \ r = [d] + 1 \ \& \ s = -[-c] - 1 \ \& \ p = s - r \to \exists_n(f(n) \ \& \ n -$ целое $\& \ a < n \ \& \ n < b) \leftrightarrow \exists_n(f(n) \ \& \ n -$ целое $\& \ a < n \ \& \ n < b \ \& \ \exists_m(m \in \{0, \dots, p\} \ \& \ n = r + m)))$

Аналогично предыдущему.

5. Свертка двух противоположных неравенств для целочисленной неизвестной в условие принадлежности конечному отрезку целых чисел.

   $\forall_{mnk}(m -$ целое $\& \ k -$ целое $\to n -$ целое $\& \ m \le n \ \& \ n \le k \leftrightarrow n \in \{m, \dots, k\})$

   Прием имеет заголовок "заменаусловия(второйтерм)" и заменяет группу условий задачи на описание. Переменная n идентифицируется с неизвестной, выражения m, k неизвестных не содержат. Другие условия задачи не содержат n. Если результат упрощения разности $k - m$ оказывается десятичной константой, прием блокируется (тогда описанные выше приемы будут выполнять разбор случаев). Уровень срабатывания равен 0.

6. Разбор случаев по условию несовпадения с данным числом.

   $\forall_{mn}(m -$ целое $\& \ n -$ целое $\to \neg(m = n) \leftrightarrow m \le n - 1 \ \vee \ n + 1 \le m)$

   Отрицание равенства является условием задачи на описание. Выражение m содержит неизвестные, n - не содержит. Второй антецедент берется в посылках, первый - обрабатывается проверочным оператором. Задача не имеет среди своих условий равенства. Заменяющая дизъюнкция сопровождается комментарием "разборслучаев". Уровень срабатывания равен 1.

7. Внесение под определяющий серию значений квантор существования двух неравенств - отрицаний альтернативных подслучаев.

$\forall_{abcfg}(\exists_n(a = f(n)$ & $n -$ целое & $g(n))\ \vee\ a < b\ \vee\ c < a \leftrightarrow \exists_n(a = f(n)$ & $n -$ целое & $g(n)$ & $b \leq f(n)$ & $f(n) \leq c)\ \vee\ a < b\ \vee\ c < a)$

Прием сужает контекст, к которому относится параметрическое описание: отсекаются случаи, охватываемые неравенствами $a < b$ и $c < a$. Обычно это позволяет получить в итоге более простую форму ответа. Дизъюнкция является условием задачи на описание. Выражение a содержит неизвестные; выражения b, c константные. Для блокировки повторного срабатывания используется комментарий "промежуток" к преобразуемому условию. Допускаются нестрогие версии неравенств $a < b, c < a$. Уровень срабатывания равен 0.

8. Существование решения неравенств.

   $\forall_{abc}(c = (0 \leq b - a) \to \exists_n(n -$ целое & $a \leq n$ & $n \leq b) \leftrightarrow 0 \leq [b] + [-a])$

   Прием имеет заголовок "связка". Исключаются все условия задачи на описание с несущественной неизвестной n. Предполагается, что эти условия идентифицируются с утверждениями под квантором существования. Если задача имела среди своих условий неравенство, выражающее невырожденность промежутка $[a, b]$, то такое неравенство становится после применения приема избыточным. Чтобы удалить его, антецедент упрощает неравенство $0 \leq b - a$ с помощью нормализатора общей стандартизации нестрогих неравенств "нормменьшеилиравно". Указатель "удалениеусловия(условие(не(заголовок(фикс(0 2) истина)))x3)" удаляет результат упрощения c из условий задачи, если только неравенство $0 \leq [b]+[-a]$ не было преобразовано в константу "истина". Уровни срабатывания равны 0 и 2.

   $\forall_{mnk}(m -$ целое & $n -$ целое & $0 \leq n - m - 1 \to \exists_a(a -$ целое & $m \leq a$ & $a \leq n$ & $\neg(a = k)))$

   Заголовок приема - "связка". Неизвестная a не входит в выражения m, n, k. Уровень срабатывания равен 1.

9. Замена дизъюнкции неравенства и равенства на неравенство.

   $\forall_{mnk}(k -$ целое & $m -$ целое & $n -$ целое & $n - m = 1 \to k = m \vee n \leq k \leftrightarrow m \leq k)$

   Прием добавляет к лучу $[n, \infty)$ точку $m = n - 1$. Первые три антецедента обрабатываются проверочными операторами, последний - выделен указателем "идентификатор". Уровень срабатывания равен 3.

10. Учет отрицания равенства.

    $\forall_{mn}(m -$ целое & $n -$ целое $\to m \leq n$ & $\neg(m = n) \leftrightarrow m \leq n - 1)$

    $\forall_{mn}(m -$ целое & $n -$ целое $\to n \leq m$ & $\neg(m = n) \leftrightarrow n + 1 \leq m)$

    От целочисленного луча отбрасывается крайняя точка. Переменная m идентифицируется с неизвестной задачи на описание, выражение n известно. Уровень срабатывания равен 1.

11. Разбор случаев для выражения, принимающего натуральные значения.

    $\forall_{xn}(x \leq n$ & $x -$ натуральное $\to \exists_m(m \in \{1, \ldots, n\}$ & $x = m))$

    Прием имеет заголовок "выводусловия". Первый антецедент идентифицируется с условием задачи на описание, причем x оказывается неизвестной, а n

- натуральной константой, не превосходящей 5. Второй антецедент обрабатывается проверочным оператором. Выводимое утверждение разворачивается в дизъюнкцию и помечается комментарием "разборслучаев". Уровень срабатывания равен 2. На той же теореме создан аналогичный прием вывода, применяемый в задачах на исследование.

12. Разрешение неравенства относительно параметра суммирования либо перемножения. Приемы выполняют приведение к стандартному виду условий на область действия конечной суммы либо конечного произведения. Напомним, что обычно эта область задается условиями "целое(i)", "$a \leq i$", "$i \leq b$".

    $\forall_{ai}(0 \leq i + a \leftrightarrow -a \leq i)$

    $\forall_{ai}(0 \leq a - i \leftrightarrow i \leq a)$

    Преобразуемое неравенство расположено внутри описателя "отображение", причем переменная i входит в сязывающую приставку данного описателя, а сам описатель является операндом операции "суммавсех" либо "произведениевсех". Уровень срабатывания равен 1. Следующий прием исключает избыточное неравенство для параметра i:

    $\forall_{abi}(a \leq i \ \& \ 0 \leq a - b \rightarrow b \leq i)$

    Первый антецедент берется в контексте, второй - обрабатывается проверочным оператором. Неравенство расположено под описателем "отображение", связывающая приставка которого содержит переменную i. Уровень срабатывания равен 0.

13. Устранение модуля.

    $\forall_n(n - \text{целое} \rightarrow 1 \leq |n| \leftrightarrow \neg(n = 0))$

    Антецедент обрабатывается проверочным оператором. Уровень срабатывания равен 1.

14. Исключение целой части.

    $\forall_{mnx}(0 < n \ \& \ m - \text{целое} \rightarrow 0 \leq [x/n] - m \leftrightarrow 0 \leq x - mn)$

    $\forall_{mnx}(0 < k \ \& \ m - \text{целое} \rightarrow 0 \leq m - [x/k] \leftrightarrow 0 < km + k - x)$

    Антецеденты обрабатываются проверочными операторами. Уровень срабатывания равен 0.

15. Переход к строгому неравенству в описании класса.

    $\forall_{fgP}(f(x) - \text{целое} \ \& \ g(x) - \text{целое} \rightarrow \text{set}_x(f(x) + 1 \leq g(x) \ \& \ P(x)) = \text{set}_x(f(x) < g(x) \ \& \ P(x)))$

    Антецеденты обрабатываются проверочными операторами. Указатели "занесениепосылки(1 значение(x40 x23))", "занесениепосылки(2 значение(x40 x23))" присоединяют к списку посылок проверочного оператора конъюнктивные члены утверждения $P(x)$. Переменные f, g, P - функциональные. Выражение $f(x)$ не является суммой, выражение $g(x)$ - переменной. Уровень срабатывания равен 2.

16. Переход к отрицанию равенства.

    $\forall_{kmnp}(k \in \{m, \ldots, n\} \ \& \ p = m + 1 \rightarrow 0 \leq k - p \leftrightarrow \neg(k = m))$

$\forall_{kmnp}(k \in \{m, \ldots, n\}\ \&\ p = n - 1 \to 0 \leq p - k \leftrightarrow \neg(k = n))$

Первый антецедент берется в контексте, второй - выделен указателем "идентификатор".

17. Использование константной оценки для целочисленной переменной.

    $\forall_{abcn}(0 \leq ab + c\ \&\ b - \text{целое}\ \&\ a \leq 0\ \&\ 0 \leq b + n \to 0 \leq 0 \leq c - an)$

    Прием выводит следствие в посылках задачи на доказательство либо на исследование. Все антецеденты, кроме третьего, идентифицируются с посылками задачи, причем b - переменная, n - целочисленная константа. Третий антецедент обрабатывается проверочным оператором. Уровень срабатывания равен 6.

18. Сопровождение ответа неравенствами для целочисленных параметров.

    $\forall_{abx}(a \leq x\ \&\ x \leq b \to a \leq b)$

    Прием имеет заголовок "выводусловия". Антецеденты идентифицируются с условиями задачи на описание, причем x - неизвестная, выражения a, b известны. Неравенство $a \leq b$ имеет не менее двух целочисленных параметров. Неизвестная x не входит в другие условия задачи, кроме, быть может, условия "x – число". Уровень срабатывания равен 0.

Учет неравенств при решении уравнений

1. Вывод неравенства для неотрицательности неизвестного дискриминанта квадратного трехчлена.

   В некоторых системах уравнений ответ получается после того, как усматриваются два встречных нестрогих неравенства для значений неизвестной. Одним из способов получения таких неравенств является вывод условия неотрицательности дискриминанта квадратного трехчлена с неизвестными коэффициентами:

   $\forall_{abcdex}(d = b^2 - 4a(c - e)\ \&\ ax^2 + bx + c = e \leftrightarrow 0 \leq d)$

   Прием имеет заголовок "вывод". Второй антецедент идентифицируется с посылкой задачи на исследование, не имеющей цели "известно". Выражение x содержит неизвестные. Дискриминант d, вычисляемый первым антецедентом, тоже содержит неизвестные. Заметим, что при его вычислении последовательно используются нормализаторы "нормплюс", "стандплюс", "уравнплюс". Таким образом, обеспечивается раскрывание скобок и приведение подобных членов с неизвестными. Коэффициент b отличен от нуля. Остаточные члены c не содержат слагаемого, делящегося на выражение x. Дополнительно проверяется выполнение следующих требований:

   (a) Число слагаемых выражения d не превосходит 3.

   (b) Либо выражения a, b не содержат неизвестных степеней с дробным показателем, либо выражение x имеет неизвестную тригонометрическую операцию.

   (c) Задача не имеет уравнения, линейного по всем своим неизвестным.

   (d) Результирующее неравенство, после обработки нормализатором решения нестрогих неравенств "уравнменьшеилиравно", имеет не более 4 дизъюнкций.

Заметим, что после упрощения результирующее неравенство может превратиться в дизъюнкцию, некоторые члены которой содержат равенства. В этом случае оно сопровождается комментарием "разборслучаев". Прием срабатывает на уровнях 5,6,7. Уровень 7 используется только в задачах с единственной неизвестной, уровень 5 - в задачах с несколькими неизвестными, имеющих уравнение, линейное по какой-либо неизвестной, и уровень 6 - в прочих задачах с несколькими неизвестными.

2. Противоположные нестрогие неравенства в дизъюнктивном члене.

   $\forall_{abcd}(a \leq b \rightarrow b \leq a \;\&\; c \;\vee\; d \leftrightarrow b = a \;\&\; c \;\vee\; d)$

   Дизъюнкция является посылкой задачи на исследование. Хотя бы одно из выражений a, b содержит неизвестные. Антецедент идентифицируется с посылкой. Прием срабатывает на уровне 2.

3. Попытка усмотреть различие знаков частей уравнения. Если усматривается, что знаки частей уравнения различны, то оно заменяется на константу "ложь":

   $\forall_{ab}(b < 0 \;\&\; 0 \leq a \rightarrow \neg(a = b))$

   $\forall_{ab}(a \leq 0 \;\&\; 0 < b \rightarrow \neg(a = b))$

   Уравнение является условием задачи на описание, причем либо отсутствует цель (известно ...), либо a, b не являются неизвестными. Введен сильный ограничитель трудоемкости. Уровень срабатывания равен 2. По первой теореме создана еще одна версия приема - для задач на описание, имеющих цель "стандравно". Такие задачи используются для решения систем уравнений, возникающих в геометрических задачах. Здесь уравнение расположено внутри дизъюнкции $a = b \vee a = c$, и успешное применение приема предотвратит разбор случаев во внешней геометрической задаче. Требуется, чтобы выражение a содержало неизвестные, но не являлось неизвестной; выражение b - было известным и отличным от 0. Уровень срабатывания тоже равен 2.

4. Попытка декомпозиции уравнения при помощи операторов "верхняяоценка", "нижняяоценка".

   Если удается оценить сверху или снизу некоторые невырожденные подвыражения уравнения, причем из соображений монотонности вытекает, что уравнение может выполняться только для найденных экстремальных значений этих подвыражений, то происходит разбиение уравнения на несколько более простых уравнений:

   $\forall_{abcde}(|a| \leq b \;\&\; |c| \leq d \;\&\; bd - e = 0 \rightarrow ac = e \leftrightarrow |a| = b \;\&\; |c| = d \;\&\; 0 \leq ac)$

   Уравнение является условием задачи на описание. Оба сомножителя a, c содержат неизвестные, выражение e - не содержит. Фактически a идентифицируется как один из сомножителей, c - как произведение оставшихся сомножителей. Первые два антецедента обращаются к синтезатору "верхняяоценка" для определения верхних оценок b, d модулей сомножителей. Третий антецедент проверяет равенство правой части e произведению найденных оценок. Введен умеренный ограничитель трудоемкости. Уровень срабатывания равен 6.

   $\forall_{abcdepx}(\sqrt{a^2 + b^2} \leq p \;\&\; e \leq c \;\&\; d + p - e = 0 \rightarrow a\cos x + b\sin x + c = d \leftrightarrow a\cos x + b\sin x = -p \;\&\; c = e)$

Выражения x, c содержат неизвестные, d - константа. Первый антецедент определяет верхнюю оценку p для квадратного корня из суммы квадратов коэффициентов при синусе и косинусе, второй - нижнюю оценку для остаточной суммы c. Третий антецедент проверяет, что d равно разности $e - p$, т.е. что уравнение может выполняться лишь для наименьшего значения e выражения c и наименьшего значения $-p$ суммы двух тригонометрических членов. Уровень срабатывания равен 4.

$\forall_{abcde}(a \leq d \ \& \ b - c \leq e \ \& \ d + e = 0 \rightarrow a + b = c \leftrightarrow a = d \ \& \ b = c + e)$

$\forall_{abcde}(d \leq a \ \&e \leq b - c \ \& \ d + e = 0 \rightarrow a + b = c \leftrightarrow a = d \ \& \ b = c + e)$

Приемы аналогичны предыдущему. Выделяется неизвестное слагаемое a левой части уравнения; b - неизвестная сумма остальных слагаемых. Правая часть c известна. Уровень срабатывания равен 6. На двух последних теоремах созданы еще две версии приемов, срабатывающие на уровне 1. В них переменная a идентифицируется как сумма всех слагаемых, содержащих неизвестную тригонометрическую операцию. При этом проверяется наличие неизвестной, входящей одновременно в a и b.

$\forall_{abcdepx}(p = b + a^2/8b \ \& \ b \leq 0 \ \& \ e \leq c \ \& \ p + e - d = 0 \rightarrow a \sin x + b \cos(2x) + c = d \leftrightarrow a \sin x + b \cos(2x) = p \ \& \ c = e)$

Выражения a, b, d константные; x, c содержат неизвестные. Здесь верхняя оценка p модуля суммы двух тригонометрических слагаемых вычисляется первым антецедентом, выделенным указателем "идентификатор". Третий антецедент находит нижнюю оценку e остаточной суммы c с помощью синтезатора "нижняяоценка". Уровень срабатывания равен 6.

$\forall_{abcde}(a \leq b \ \& \ c \leq d \ \& \ bd - e = 0 \ \& \ b < 0 \ \& \ d < 0 \rightarrow ac = e \leftrightarrow a = b \ \& \ c = d)$

Сомножитель a и остаточное произведение c содержат неизвестные, e известно. Уровень срабатывания равен 6.

$\forall_{abcdep}(a \leq b \ \& \ c \leq d \ \& \ bd - e = 0 \ \& \ ac = p \rightarrow p = e \leftrightarrow a = b \ \& \ c = d \ \vee \ a \leq 0 \ \& \ c \leq 0 \ \& \ p = e)$

Четвертый антецедент, выделенный указателем "идентификатор", определяет разбиение произведения p на сомножитель a и остаточное произведение c. Затем первые два антецедента, обращаясь к синтезаторам, находят верхние оценки b, d выражений a, c. Третий антецедент проверяет, что их произведение равно e. Прием выводит дизъюнкцию, которая в одном подслучае выполняет декомпозицию уравнения, а в другом - указывает дополнительные условия неположительности сомножителей. Используются ускоряющие фильтры, проверяющие, что не усматривается ни одно из неравенств $e < 0$, $a \leq 0$, $c \leq 0$. Последние две проверки нужны также для предотвращения зацикливания. Уровень срабатывания равен 8.

5. Попытка усмотреть ложность уравнения при помощи операторов "верхняяоценка", "нижняяоценка".

   $\forall_{abc}(c \leq a \ \& \ 0 < c - b \rightarrow \neg(a = b))$

   Прием имеет заголовок "второйтерм". Он применяется к уравнениям с тригонометрической операцией "синус" либо "косинус". Текущая задача - на описание. Выражение a содержит неизвестные, b - не содержит. Первый антецедент определяет c с помощью синтезатора "нижняяоценка", второй антецедент обрабатывается проверочным оператором. Введен умеренный ограничитель трудоемкости. Уровень срабатывания равен 5.

$\forall_{abcdepx}(\sqrt{a^2+b^2} \leq p \;\&\; 0 < d - p \rightarrow a\cos x + b\sin x = d \leftrightarrow \text{ложь})$

Хотя бы один из коэффициентов a, b содержит неизвестные, d - константное выражение. Первый антецедент обрабатывается синтезатором "верхняяоценка", второй - проверочным оператором. Уровень срабатывания равен 4.

6. Замена серии значений параметра, ограниченного сверху и снизу, на конечное множество его значений.

   $\forall_{fPQabpq}(0 < a \;\&\; 0 < p \;\&\; (\exists_n(n - \text{целое} \;\&\; x = f(n) \;\&\; P(n)) \;\&\; -b/a \leq x \;\&\; x \leq q/p) = Q(x) \rightarrow \exists_n(n-\text{целое} \;\&\; x = f(n) \;\&\; P(n)) \;\&\; 0 \leq ax+b \;\&\; 0 \leq q-px \leftrightarrow Q(x))$

   Прием применяется к тройке не содержащих неизвестных условий задачи на описание: $\exists_n(n - \text{целое} \;\&\; x = f(n) \;\&\; P(n))$, $0 \leq ax + b$, $0 \leq q - px$. Первое условие определяет серию значений известного параметра x, второе и третье - ограничивают область изменения этого параметра конечным промежутком. Выражения a, b, p, q не содержат параметра x. Третий антецедент, выделенный указателем "идентификатор", разрешает с помощью вспомогательной задачи на описание выделенные условия относительно переменной x. Проверяется, что каждый дизъюнктивный член результата $Q(x)$ имеет вид равенства, определяющего значение x. Указатели "альтернатива" допускают случаи строгих неравенств. Уровень срабатывания равен 3.

Подбор примера

$\forall_{ax}(a - \text{число} \;\&\; x = a \rightarrow x \leq a)$

Прием подбирает пример a значения неизвестной x в задаче на описание, не имеющей других условий с x, кроме условия "$x \leq a$" и, быть может, условия "x- число". Задача имеет цель "пример". Выражение a известно. Отсутствует цель "независит ...", запрещающая зависимость неизвестной x от параметра выражения a. Первый антецедент обрабатывается проверочным оператором, второй - определяет заменяющее условие для попытки подбора значения. Указатель "дробь" обеспечивает применение приема при перестановке частей неравенства. Уровень срабатывания равен 1.

Группировка посылок - нестрогого неравенства и отрицания равенства

$\forall_{a}(a - \text{число} \rightarrow \neg(a = 0) \;\&\; 0 \leq a \leftrightarrow 0 < a)$

$\forall_{ab}(\neg(a = b) \;\&\; a \leq b \leftrightarrow a < b)$

$\forall_{ab}(\neg(a - b = 0) \;\&\; b - a \leq 0 \leftrightarrow 0 < a - b)$

$\forall_{ab}(\neg(a - b = 0) \;\&\; 0 \leq b - a \leftrightarrow a - b < 0)$

Приемы имеют заголовок "заменатермов(второйтерм)" и заменяют две посылки задачи на одно строгое неравенство. Уровень срабатывания равен 2.

Группировка условий - нестрогого неравенства и отрицания равенства

$\forall_{ab}(\neg(a = b) \;\&\; a \leq b \leftrightarrow a < b)$

Прием имеет заголовок "заменаусловия(второйтерм)" и применяется к паре условий задачи на описание. Уровни срабатывания равны 1,3 и 5.

$\forall_{ab}(\neg(a-b=0)\ \&\ b-a\leq 0\leftrightarrow 0<a-b)$

$\forall_{ab}(\neg(a-b=0)\ \&\ 0\leq b-a\leftrightarrow a-b<0)$

Аналогично предыдущему, но уровни срабатывания равны 2,4 и 6.

Обратный вывод в задаче с целью "независит"

$\forall_{abc}(0\leq a-b\ \&\ c=a\rightarrow b\leq c)$

Прием имеет заголовок "подборзначений". Неравенство $b\leq c$ идентифицируется с условием задачи на описание, имеющей цель (независит A). При этом c - неизвестная, b - некоторое выражение, содержащее переменные списка A, т.е. такие переменные, от которых не должен зависеть ответ. Первый антецедент берется в посылках, причем выражение a не содержит переменных списка A. Второй антецедент, выделенный указателем "подборзначений", представляет собой заменяющее условие при попытке подбора значения неизвестной c. Уровень срабатывания равен 3.

Вывод следствий

1. Усмотрение неотрицательности слагаемого.

   $\forall_{ab}(b\leq 0\ \&\ 0\leq a+b\rightarrow 0\leq a)$

   Второй антецедент идентифицируется с посылкой задачи на описание, имеющей цель (известно ...). Эта посылка не имеет невырожденных числовых атомов. a - слагаемое ее правой части, причем все оставшиеся слагаемые имеют заголовок "минус". Первый антецедент обрабатывается проверочным оператором. Введен сильный ограничитель трудоемкости. Уровень срабатывания равен 0.

2. Линейная комбинация неравенств в задачах на описание, имеющих цель "независит".

   $\forall_{abcdef}(0\leq a+bc\ \&\ 0\leq d+ef\ \&\ b<0\ \&\ e<0\rightarrow bec+bef+ae+bd\leq 0)$

   $\forall_{abcdef}(0\leq a+bc\ \&\ 0\leq d+ef\ \&\ 0<b\ \&\ 0<e\rightarrow 0\leq bec+bef+ae+bd)$

   Идентификация начинается с усмотрения в неравенстве задачи на описание, имеющей цель (независит A), суммы $c+f$, не расположенной под модулем. Первые два антецедента идентифицируются после этого с посылками задачи. Выражения c,f содержат переменные списка A, выражения a,b,d,e - не содержат. Третий и четвертый антецеденты обрабатываются проверочными операторами. Новая посылка представляет собой неравенство для суммы $c+f$, домноженной на коэффициент be. Оно может оказаться полезным при прямом либо обратном выводе, исключающем зависимость от переменных списка A. Уровень срабатывания равен 4.

3. Специальные кванторные неравенства.

   $\forall_{fghpA}(\forall_x(A(x)\rightarrow f(x)\leq g(x))\ \&\ \forall_y(A(y)\rightarrow h(y)\leq g(y))\ \&\ \forall_z(A(z)\rightarrow p(z)\leq f(z))\ \&\ \forall_v(A(v)\rightarrow p(v)\leq h(v))\rightarrow\forall_x(A(x)\rightarrow 0\leq g(x)-p(x)-|f(x)-h(x)|))$

   Антецеденты идентифицируются с посылками задачи. Они означают, что величины $h(x),f(x)$ для любого x, удовлетворяющего условию $A(x)$, заключены между $p(x)$ и $g(x)$. Переменные f, g, h, p - функциональные. Допускаются задачи на доказательство, исследование, а также задачи на описание, имеющие цель "пример". Выводимое следствие выражает тот факт, что модуль разности $f(x)$ и $h(x)$ не превосходит $g(x) - p(x)$. Уровень срабатывания равен 3.

$\forall_f(0 \leq f(n)$ & $\forall_{mn}(m -$ натуральное & $n -$ натуральное $\rightarrow f(m+n) \leq f(m) + f(n)) \rightarrow \forall_{mn}(m -$ натуральное & $n -$ натуральное $\rightarrow f(m) \leq [m/n]f(n) + (0, \text{если } m|n, \text{иначе } f(m(\text{mod } n)))))$

Второй антецедент берется в посылках, первый - обрабатывается проверочным оператором, которому передается дополнительная посылка "$n -$ натуральное". Прием применяется в задачах на описание, имеющих цель "длялюбого". Уровень срабатывания равен 3.

Попытка опровергнуть неравенство при помощи оператора "провменьше"

Для усмотрения ложности нестрогого неравенства можно использовать усиленный оператор усмотрения истинности строгих неравенств (т.е. оператор "провменьше"):

$\forall_{ab}(a < b \rightarrow \neg(b \leq a))$

Неравенство $b \leq a$ является посылкой задачи на исследование, сопровождаемой комментарием "вспомописание". Уровень срабатывания равен 5.

Выдача ответа задачи на описание

Используются следующие теоремы приемов, определяющие шаблоны ответов:

"явное(x1 набор(число(x1)меньше(x1 x2)меньшеилиравно(x3 x1))набор(не(равно(x1 x4)))пустоеслово)",

"явное(x1 набор(число(x1)меньшеилиравно(x1 x2)меньше(x3 x1)) набор(не(равно(x1 x4)))пустоеслово)",

"явное(x1 набор(число(x1)меньшеилиравно(x1 x2)меньшеилиравно(x3 x1))набор(не(равно(x1 x4)))пустоеслово)",

"явное(x14 набор(натуральное(x14)меньшеилиравно(x14 x1)меньшеилиравно(x2 x14)) пустоеслово набор(меньшеилиравно(x2 x1)))",

"явное(x14 набор(целое(x14)меньшеилиравно(x14 x1)меньшеилиравно(x2 x14)) пустоеслово набор(меньшеилиравно(x2 x1)))"

Приемы с этими теоремами аналогичны случаю строгих неравенств.

Усмотрение неотрицательности параметра, явно определенного равенством

В задачах на исследование, имеющих цель "известно", целесообразно сразу выводить посылку, указывающую на неотрицательность числового параметра - иначе его неотрицательность по ходу решения может стать неочевидной. Данную работу выполняют следующие два приема.

$\forall_{abc}(ab = c$ & $0 < b$ & $0 \leq c \rightarrow 0 \leq a)$

$\forall_{ab}(a = b$ & $0 \leq b \rightarrow 0 \leq a)$

В обоих случаях a идентифицируется с переменной. Первый антецедент - посылка задачи на исследование, имеющей цель "известно". Остальные антецеденты обрабатываются проверочными операторами. У первого приема a является известным параметром, не входящим в b, причем правая часть c имеет заголовок "расстояние", либо "угол", либо "площадь". У второго приема либо переменная a - параметр, либо b - дробное выражение. В обоих случаях b должно иметь некорневое вхождение одного из символов "расстояние", "угол", "площадь", периметр". Требуется также, чтобы отсутствовала посылка вида $a = t$, где t имеет заголовком один из указанных символов. Первый прием срабатывает на уровне 2, второй - на уровне 1.

Объединение случаев с противоположными неравенствами

$\forall_{ab}(0 \leq a \,\&\, b \leq 0 \;\vee\; a \leq 0 \,\&\, 0 \leq b \leftrightarrow ab \leq 0)$

$\forall_{ab}(0 \leq a \,\&\, 0 \leq b \;\vee\; a \leq 0 \,\&\, b \leq 0 \leftrightarrow 0 \leq ab)$

$\forall_{ab}(\neg(a = 0) \rightarrow a < 0 \,\&\, b \leq 0 \;\vee\; 0 < a \,\&\, 0 \leq b \leftrightarrow 0 \leq ab)$

$\forall_{ab}(\neg(a = 0) \rightarrow a < 0 \,\&\, 0 \leq b \;\vee\; 0 < a \,\&\, b \leq 0 \leftrightarrow ab \leq 0)$

Приемы применяются к дизъюнкции, являющейся условием задачи на описание. Эта задача имеет цели "редакция", "и", т.е. ответ должен иметь вид конъюнкции элементарных утверждений. Дизъюнкция не содержит неизвестных. Антецеденты обрабатываются проверочным оператором. Уровень срабатывания равен 2.

Проверочный оператор "усмменьшеилиравно"

Оператор усматривает истинность нестрогих неравенств. В тех случаях, когда его приемы аналогичны приемам оператора "усмменьше", ограничиваемся лишь названием пункта.

1. Непосредственное усмотрение.

2. Переход к неравенству с нулевой частью.

3. Устранение минуса.

4. Усмотрение знака суммы путем установления знаков слагаемых.

   $\forall_{ab}(a \leq 0 \,\&\, b \leq 0 \rightarrow a + b \leq 0)$

   $\forall_{ab}(0 \leq a \,\&\, 0 \leq b \rightarrow 0 \leq a + b)$

   В отличие от строгих неравенств, появляется возможность одновременной обработки всех слагаемых вне зависимости от их числа. Она обеспечивается использованием указателя "дистрибразвертка(...)". Неравенство должно быть неконстантным. Уровень срабатывания равен 2.

5. Усмотрение знака произведения.

   $\forall_{ab}(a \leq 0 \,\&\, 0 \leq b \rightarrow ab \leq 0)$

   $\forall_{ab}(a \leq 0 \,\&\, 0 \leq b \rightarrow ab \leq 0)$

   $\forall_{ab}(0 \leq a \,\&\, 0 \leq b \rightarrow 0 \leq ab)$

   Последний прием создан в двух вариантах, Первый вариант снабжен указателем "дистрибразвертка" для усмотрения одновременной неотрицательности всех сомножителей. Второй не имеет данного указателя, у него первый антецедент берется в посылках и a является произведением. Приемы для первых двух теорем и первый вариант приема третьей теоремы имеют уровень срабатывания 2, второй вариант - уровень срабатывания 3. Предусмотрен еще один прием, позволяющий отбрасывать строго положительные множители и блокировать после этого альтернативные попытки:

$\forall_{ab}(0 < a \;\&\; 0 \leq b \rightarrow 0 \leq ab)$

Оба антецедента обрабатываются проверочными операторами. Блокировку альтернативных попыток после проверки первого антецедента обеспечивает указатель "спуск(1)". Прием применяется только в контекстах, содержащих указание о стремлении параметра к некоторому "предельному" значению. Уровень срабатывания равен 2.

6. Устранение дроби.

   $\forall_{ab}(ab \leq 0 \rightarrow a/b \leq 0)$

   $\forall_{ab}(0 \leq ab \rightarrow 0 \leq a/b)$

   Приемы снабжены указателем "спуск", блокирующим альтернативные попытки. Антецедент обрабатывается проверочным оператором. Если дробь неконстантная, уровень обращения равен 1. Иначе он равен 4 (для этой ситуации созданы отдельные версии приемов).

7. Устранение степени.

   $\forall_{ab}(0 \leq a \rightarrow 0 \leq a^b)$

   $\forall_{ab}(\text{числитель}(b) - \text{even} \;\&\; b - \text{rational} \rightarrow 0 \leq a^b)$

   $\forall_{ab}(b - \text{rational} \;\&\; \neg(\text{знаменатель}(b) - \text{even}) \;\&\; \neg(\text{числитель}(b) - \text{even}) \;\&\; a \leq 0 \rightarrow a^b \leq 0)$

   Первый и последний приемы снабжены указателем "спуск", блокирующим альтернативные попытки. Уровень срабатывания равен 1. Для специальных случаев создан прием, усматривающий неотрицательность там, где она необходима по о.д.з., но явно из контекста не выводится:

   $\forall_{ab}(\text{знаменатель}(b) - \text{even} \;\&\; b - \text{rational} \rightarrow 0 \leq a^b)$

   Для срабатывания необходимо наличие комментария "стандинтеграл". Уровень срабатывания равен 1.

8. Разность степеней с одинаковыми показателями.

   $\forall_{abc}(0 < c \;\&\; 0 \leq a \;\&\; 0 \leq b \;\&\; 0 \leq a - b \rightarrow 0 \leq a^c - b^c)$

   $\forall_{abc}(0 < c \;\&\; 0 \leq a \;\&\; 0 \leq b \;\&\; a - b \leq 0 \rightarrow a^c - b^c \leq 0)$

   $\forall_{abc}(c - \text{rational} \;\&\; 0 < c \;\&\; \neg(\text{числитель}(c) - \text{even}) \;\&\; 0 \leq a - b \rightarrow 0 \leq a^c - b^c)$

   $\forall_{abc}(c - \text{rational} \;\&\; 0 < c \;\&\; \neg(\text{числитель}(c) - \text{even}) \;\&\; a - b \leq 0 \rightarrow a^c - b^c \leq 0)$

   Эти приемы аналогичны случаю строгих неравенств. Уровень срабатывания их равен 2.

   $\forall_{abcdn}(0 \leq c \;\&\; 0 \leq d \;\&\; 0 < n \;\&\; 0 < a \;\&\; 0 < b \;\&\; 0 \leq a^{1/n}c - b^{1/n}d \rightarrow 0 \leq ac^n - bd^n)$

   $\forall_{abcdn}(0 \leq c \;\&\; 0 \leq d \;\&\; 0 < n \;\&\; 0 < a \;\&\; 0 < b \;\&\; a^{1/n}c - b^{1/n}d \leq 0 \rightarrow ac^n - bd^n \leq 0)$

Выражения a, b, n идентифицируются с константами, причем n не имеет вида дроби. Антецеденты обрабатываются проверочными операторами. Уровень срабатывания равен 3.

9. Квадратный трехчлен с неположительным дискриминантом.

   $\forall_{abcde}(0 < a \ \& \ e = b^2 - 4ac \ \& \ e \leq 0 \rightarrow 0 \leq ad^2 + bd + c)$

   $\forall_{abcde}(a < 0 \ \& \ e = b^2 - 4ac \ \& \ e \leq 0 \rightarrow ad^2 + bd + c \leq 0)$

   Переменные a, b, c идентифицируются с десятичными константами. Антецеденты выделены указателем "программа". Уровень срабатывания равен 2.

10. Неравенства с радикалами.

    $\forall_{ab}(0 \leq b \rightarrow 0 \leq \sqrt{a^2 + b} - a)$

    $\forall_{ab}(0 \leq b \rightarrow 0 \leq \sqrt{a^2 + b} + a)$

    $\forall_{ab}(b \leq 0 \rightarrow \sqrt{a^2 + b} - a \leq 0)$

    Уровень срабатывания равен 1.

    $\forall_{ab}(\sqrt{a - b}\sqrt{a + b} - a^2 \leq 0)$

    $\forall_{ab}(0 \leq a \ \& \ 0 \leq b \rightarrow 0 \leq a + b - \sqrt{a^2 + b^2})$

    $\forall_{abcd}(0 \leq a \ \& \ 0 \leq c \ \& \ 0 \leq a^2 b - c^2 d \rightarrow 0 \leq a\sqrt{b} - c\sqrt{d})$

    $\forall_{abc}(0 < a \ \& \ 0 < c \rightarrow 0 \leq a - \sqrt{ac + b}\sqrt{ac - b}/c)$

    Уровень срабатывания равен 3.

    $\forall_{ab}(a^2 - b \leq 0 \rightarrow a - \sqrt{b} \leq 0)$

    Антецедент обрабатывается проверочным оператором, причем в левой части неравенства раскрываются скобки. Необходимо наличие комментария "контроль". Уровень срабатывания равен 4.

11. Слагаемое - степень минус единицы.

    $\forall_{abc}(0 \leq a - |c| \rightarrow 0 \leq a + c(-1)^b)$

    Антецедент обрабатывается проверочным оператором. Уровень срабатывания равен 2.

12. Попытка одновременного изменения знака у всех слагаемых. Этот и следующий пункты аналогичны случаю строгих неравенств.

13. Выделение суммы константных слагаемых.

14. Попытка отбрасывания слагаемых с заданным знаком.

    $\forall_{ab}(0 \leq a \ \& \ 0 \leq b \rightarrow 0 \leq a + b)$

    В отличие от приведенного выше приема, предпринимающего попытку усмотрения неотрицательности каждого слагаемого, здесь находится какое-то одно неотрицательное слагаемое a и предпринимается попытка усмотрения неотрицательности оставшейся суммы b. Прием применяется только к неконстантным неравенствам, причем необходимо наличие комментария "плюс". Уровень срабатывания равен 3.

15. Вычитание дроби из единицы.

    $\forall_{abc}(0 < a \ \& \ 0 < b \ \& \ 0 < c \ \& \ 0 \leq b - a \rightarrow 0 \leq 1 - a^c/b^c)$

    Допускается обращение c в единицу. Антецеденты обрабатываются проверочными операторами. Введен сильный ограничитель трудоемкости. Уровень срабатывания равен 3.

16. Разность модулей.

    $\forall_{ab}(0 \leq a^2 - b^2 \rightarrow 0 \leq |a| - |b|)$

    Антецедент обрабатывается проверочным оператором. Введен сравнительно сильный ограничитель трудоемкости. Уровень срабатывания равен 3.

17. Сумма с экспонентой.

    $\forall_{abcd}(1 < b \ \& \ 0 < a \ \& \ 0 \leq a + d \ \& \ 0 \leq c \rightarrow 0 \leq ab^c + d)$

    Выражения a, b, d константные, выражение c - неконстантное. Антецеденты обрабатываются проверочными операторами. Введен сильный ограничитель трудоемкости. Уровень срабатывания равен 1.

18. Разность максимума и минимума.

    $\forall_{ab}(0 \leq \max(a, b) - \min(a, b))$

    Уровень срабатывания равен 2.

19. Разность числа и минимума.

    $\forall_{ab}(0 \leq a - \min(a, b))$

    Уровень срабатывания равен 1.

20. Разность максимума и числа.

    $\forall_{ab}(0 \leq \max(a, b) - a)$

    Уровень срабатывания равен 1.

21. Простейшие случаи использования посылок.

    $\forall_{ab}(a < b \rightarrow 0 \leq b - a)$

    $\forall_{ab}(-a < b \rightarrow 0 \leq a + b)$

    Антецедент берется в посылках, причем допускается случай нестрогого неравенства. Уровень срабатывания равен 3.

    $\forall_{ab}(0 \leq a \ \& \ a < b \rightarrow 0 \leq b)$

    Второй антецедент берется в посылках, причем a - отличное от 0 константное выражение. Допускается нестрогое неравенство. Первый антецедент обрабатывается проверочным оператором. Уровень срабатывания равен 3.

    $\forall_{abc}(a \leq b \ \& \ 0 \leq c \rightarrow 0 \leq b - a + c)$

    Первый антецедент берется в посылках, второй - обрабатывается проверочным оператором. Выражение b ненулевое. Уровень срабатывания равен 3.

    $\forall_{abcx}(x \leq c \ \& \ a \leq 0 \ \& \ 0 \leq ac + b \rightarrow 0 \leq ax + b)$

    $\forall_{abcx}(c \leq x \ \& \ 0 \leq a \ \& \ 0 \leq ac + b \rightarrow 0 \leq ax + b)$

Выражения a, b, c константные, x - переменная. Выражение c ненулевое. Первый антецедент берется в посылках, остальные - обрабатываются проверочными операторами. Уровень срабатывания равен 3.

$\forall_{abc}(0 \le (a-b)c \to (b-a)c \le 0)$

Антецедент берется в посылках. Уровень срабатывания равен 1.

22. Транзитивность с использованием неравенства из посылок, имеющего нулевую часть.

    $\forall_{ab}(a+b \le 0 \mathbin{\&} 0 \le b \to a \le 0)$

    $\forall_{ab}(0 \le a+b \mathbin{\&} b \le 0 \to 0 \le a)$

    Первый антецедент берется в посылках, причем допускается случай строгого неравенства. Второй антецедент обрабатывается проверочным оператором. Выражение a должно быть неконстантным. Введен средний ограничитель трудоемкости. Уровень обращения равен 2.

    $\forall_{ab}(b \le 0 \mathbin{\&} 0 \le b-a \to a \le 0)$

    Аналогично предыдущему, но в посылках берется второй антецедент.

    $\forall_{ab}(0 \le a \mathbin{\&} 0 \le b \to 0 \le a+b)$

    Первый антецедент берется в посылках, второй - обрабатывается проверочным оператором. Выражение a неконстантное. Уровень срабатывания равен 4.

23. Транзитивность с использованием неравенства из посылок, не имеющего нулевой части.

    $\forall_{abc}(c < -a \mathbin{\&} b-c \le 0 \to a+b \le 0)$

    $\forall_{abc}(a \le c \mathbin{\&} 0 \le a+b \to 0 \le b+c)$

    $\forall_{abc}(a \le c \mathbin{\&} 0 \le b-c \to 0 \le b-a)$

    $\forall_{abc}(a < c \mathbin{\&} b+c \le 0 \to a+b \le 0)$

    $\forall_{abc}(c \le b \mathbin{\&} 0 \le c-a \to a-b \le 0)$

    Первый антецедент берется в списке посылок, причем у него допускаются как строгое, так и нестрогое неравенства. Второй антецедент обрабатывается проверочным оператором. Переменная первого антецедента, общая с проверяемым неравенством, идентифицируется с неконстантным выражением. Переменная проверяемого неравенства, не входящая в первый антецедент, может идентифицироваться с нулем. Уровень срабатывания приемов равен 2.

    $\forall_{abc}(a \le c \mathbin{\&} 0 \le b-c \to 0 \le b-a)$

    Выражения b, c константные. Первый антецедент берется в посылках, второй - обрабатывается проверочным оператором. Уровень срабатывания равен 1.

    $\forall_{abcdp}(b < p \mathbin{\&} a+cp/d \le 0 \mathbin{\&} 0 \le d \mathbin{\&} 0 \le c \to a+bc/d \le 0)$

    Выражения a, c, d, p константные, выражение b имеет невырожденный числовой атом. Первый антецедент берется в посылках, остальные обрабатываются проверочными операторами. Выражения c, d могут обращаться в единицу. Уровень срабатывания равен 4.

    $\forall_{abcd}(c+d \le a \mathbin{\&} 0 \le b+c+d \to 0 \le a+b)$

Выражения a, c суть переменные, выражения b, d константные. Первый антецедент берется в посылках, причем допускается случай строгого неравенства. Второй антецедент обрабатывается проверочным оператором. d может обращаться в 0. Уровень срабатывания равен 4.

24. Усмотрение в посылках строгой версии неравенства.

    $\forall_{ab}(a < b \to a \leq b)$

    Антецедент берется в посылках. Проверяемое неравенство неконстантное. Уровень срабатывания равен 1.

25. Усмотрение знака суммы по транзитивности.

    $\forall_{abc}(b + c \leq 0 \ \& \ 0 \leq a + c \to 0 \leq a - b)$

    $\forall_{abc}(0 \leq a + b \ \& \ 0 \leq c - b \to 0 \leq a + c)$

    $\forall_{abc}(a + c \leq 0 \ \& \ 0 \leq c - b \to a + b \leq 0)$

    Первый антецедент берется в посылках, второй - обрабатывается проверочным оператором. Для последней теоремы создана также версия приема, у которой второй антецедент берется в посылках, а первый обрабатывается проверочным оператором. Во всех случаях неравенство, извлекаемое из посылок, может быть строгим. Остаточные суммы проверяемого неравенства и неравенства из посылок либо имеют не более одного слагаемого каждая, либо первая из них константная, либо вторая константная, а первая имеет не более одного неконстантного слагаемого. Введен средний ограничитель трудоемкости. Уровень срабатывания равен 2.

    $\forall_{abc}(0 \leq a + b \ \& \ b \leq 0 \to 0 \leq a)$

    Первый антецедент берется в посылках, второй обрабатывается проверочным оператором. Переменная a идентифицируется с неконстантной суммой, не имеющей общих параметров с b. Уровень срабатывания равен 1.

    $\forall_{ab}(0 < a \ \& \ 0 \leq b \to 0 \leq a + b)$

    Первый антецедент берется в посылках, второй - обрабатывается проверочным оператором. Переменная a идентифицируется с суммой. Уровень срабатывания равен 3.

26. Усмотрение знака суммы при помощи линейных комбинаций с посылками.

    $\forall_{abcde}(b + ce \leq 0 \ \& \ 0 \leq e(ae - bd) \ \& \ ed < 0 \to 0 \leq a + cd)$

    $\forall_{abcde}(0 \leq b + ce \ \& \ 0 \leq e(ae - bd) \ \& \ 0 < ed \to 0 \leq a + cd)$

    $\forall_{abcde}(0 \leq b + ce \ \& \ e(ae - bd) \leq 0 \ \& \ ed < 0 \to a + cd \leq 0)$

    $\forall_{abcde}(b + ce \leq 0 \ \& \ e(ae - bd) \leq 0 \ \& \ 0 < ed \to a + cd \leq 0)$

    Первый антецедент берется в посылках, причем допускается идентификация его со строгим неравенством. Остальные антецеденты обрабатываются проверочными операторами. Выражения a, b, d, e константные, c - неконстантное и идентифицируется как пересечение групп сомножителей. Уровень срабатывания равен 2.

    $\forall_{abcx}(0 < a \ \& \ c \leq x \ \& \ 0 \leq ac + b \to 0 \leq ax + b)$

    $\forall_{abcx}(a < 0 \ \& \ x \leq c \ \& \ 0 \leq ac + b \to 0 \leq ax + b)$

$\forall_{abcx}(0 < a \ \& \ x \leq c \ \& \ ac + b \leq 0 \to ax + b \leq 0)$

$\forall_{abcx}(a < 0 \ \& \ c \leq x \ \& \ ac + b \leq 0 \to ax + b \leq 0)$

Второй антецедент берется в посылках, причем допускается случай строгого неравенства, первый и третий - обрабатываются проверочными операторами. Выражения a, b константные, x - переменная. Приемы включаются при наличии комментария "нижняяоценка". Уровень срабатывания равен 4.

$\forall_{abcx}(x \leq c \ \& \ b < 0 \ \& \ 0 \leq a + bc \to 0 \leq a + bx)$

Первый антецедент берется в посылках, второй и третий - обрабатываются проверочными операторами. Выражения a, b, c константные, причем c не равно 0. Прием включается при наличии комментария "проверка". Уровень срабатывания равен 4.

$\forall_{pqrsxy}(0 \leq px + qy \ \& \ qs < 0 \ \& \ (rq - ps)xq \leq 0 \to rx + sy \leq 0)$

$\forall_{pqrsxy}(0 \leq px + qy \ \& \ 0 < qs \ \& \ 0 \leq (rq - ps)xq \to 0 \leq rx + sy)$

Первый антецедент берется в списке посылок, причем допускается случай строгого неравенства. Второй и третий антецеденты обрабатываются проверочным оператором. Выражения p, q, r, s константные, выражения x, y - не константные и идентифицируются по пересечению групп сомножителей. Уровень срабатывания равен 3.

27. Использование неравенств с радикалами.

    $\forall_{ab}(0 \leq \sqrt{(a - b)(a + b)} + a \to 0 \leq a)$

    Уровень срабатывания равен 3.

    $\forall_{ab}(-\sqrt{a} \leq b \ \& \ b \leq \sqrt{a} \to 0 \leq a - b^2)$

    Оба антецедента берутся в посылках. Выражение b неконстантное. Уровень срабатывания равен 4.

28. Усмотрение неравенств с модулем.

    $\forall_{ab}(a + b \leq 0 \to b - |a| \leq 0)$

    $\forall_{ab}(0 \leq a + b \ \& \ 0 \leq a - b \to 0 \leq a - |b|)$

    Антецеденты берутся в посылках, причем допускаются случаи строгих неравенств. Уровень срабатывания равен 3.

29. Неравенство для разности квадратов.

    $\forall_{a}(0 \leq 1 - a^2 \to 0 \leq 1 - a)$

    $\forall_{a}(0 \leq 1 - a^2 \to 0 \leq 1 + a)$

    $\forall_{abc}(a \leq b\sqrt{c} \ \& \ -b\sqrt{c} \leq a \to 0 \leq b^2c - a^2)$

    Антецеденты берутся в посылках, уровень срабатывания равен 3.

    $\forall_{abc}(0 \leq a \ \& \ 0 \leq b \ \& \ 0 \leq a^2 - b^2c \to 0 \leq a - b\sqrt{c})$

    Третий антецедент берется в посылках, первые два - обрабатываются проверочными операторами. Уровень срабатывания равен 3.

    $\forall_{abc}(0 \leq a^2 - b^2 + c \ \& \ c \leq 0 \to 0 \leq (a - b)(a + b))$

Первый антецедент берется в посылках, причем допускается случай строгого неравенства. Второй антецедент обрабатывается проверочным оператором. Уровень срабатывания равен 2.

$\forall_{ab}(0 \leq (a-b)(a+b) \to 0 \leq a^2 - b^2)$

Антецедент берется в посылках. Уровень срабатывания равен 3.

30. Использование кванторных посылок.

    $\forall_{afg}(f(a) \ \& \ \forall_x(f(x) \to 0 < g(x)) \to 0 \leq g(a))$

    Второй антецедент берется в посылках, причем допускается случай нестрогого неравенства. Переменная f функциональная, переменная g - обычная. Первый антецедент выделен указателем "очевидно", т.е. будет обрабатываться проверочным оператором, определяемым конкретным видом утверждения $f(a)$. Указатель "удалениепосылок(1 x2 заголовок(x2 длялюбого))" отбрасывает при последней проверке все посылки, имеющие вид кванторных импликаций. Уровень обращения равен 3.

    $\forall_{apqr}(p(a) \ \& \ \forall_i(p(i) \to q(i) \leq r(i)) \to 0 \leq r(a) - q(a))$

    Второй антецедент берется в посылках. Переменная p функциональная, переменные q, r - обычные. Первый антецедент выделен указателем "очевидно". Уровень срабатывания равен 3.

31. Домножение на знаменатель слагаемого из неравенства в посылках.

    $\forall_{abcd}(0 < c \ \& \ 0 \leq a + b/c \ \& \ 0 \leq d - b \to 0 \leq ac + d)$

    Второй антецедент берется в посылках, причем допускается строгое неравенство. Остальные антецеденты обрабатываются проверочными операторами. Выражение c представляет собой десятичную константу, выражение a неконстантное. Введен сильный ограничитель трудоемкости. Уровень срабатывания приема равен 3.

32. Отбрасывание степени синуса либо косинуса.

    $\forall_{abcn}(0 \leq a + b \ \& \ b \leq 0 \ \& \ 0 \leq n \to 0 \leq a + b(\sin c)^n)$

    $\forall_{abcn}(0 \leq a + b \ \& \ b \leq 0 \ \& \ 0 \leq n \to 0 \leq a + b(\cos c)^n)$

    Первый антецедент берется в посылках, остальные - обрабатываются проверочными операторами. Для ускорения проверок сразу отсекаются случаи единичного значения b. Уровень срабатывания равен 4.

33. Оценка для квадрата.

    $\forall_{abx}(0 < x - a \ \& \ 0 < a \ \& \ 0 \leq a^2 + b \to 0 \leq x^2 + b)$

    Первый антецедент берется в посылках, остальные - обрабатываются проверочными операторами. Выражения a, b константные, x - неконстантное. Уровень срабатывания равен 4.

34. Неравенства для наименьшего и наибольшего элементов числового множества.

    $\forall_{abc}(b \subseteq \mathbf{R} \ \& \ c \in b \ \& \ \text{наименьший}(a,b) \to a \leq c)$

    $\forall_{abc}(b \subseteq \mathbf{R} \ \& \ c \in b \ \& \ \text{наибольший}(a,b) \to \ \leq a)$

Последний антецедент берется в посылках, первые два - обрабатываются проверочными операторами. Выражение a неконстантное. Уровень срабатывания равен 3.

35. Неравенства для нижней и верхней граней числового множества.

$\forall_{abc}(a \subseteq \mathbf{R}\ \&\ c \in a\ \&\ \text{нижняягрань}(b, a) \to b \le c)$

$\forall_{abc}(a \subseteq \mathbf{R}\ \&\ c \in a\ \&\ \text{верхняягрань}(b, a) \to\ \le b)$

Последний антецедент берется в посылках, первые два - обрабатываются проверочными операторами. Выражение c неконстантное. Уровень срабатывания равен 3.

$\forall_{abcd}(0 \le ad + c\ \&\ 0 < a\ \&\ \text{нижняягрань}(d, b) \to 0 \le a \inf b + c)$

$\forall_{abcd}(ad + c \le 0\ \&\ 0 < a\ \&\ \text{верхняягрань}(d, b) \to a \sup b + c \le 0)$

Аналогично предыдущему, но без требований неконстантности.

36. Использование равенства. Приводимые ниже приемы, как правило, ориентированы на специальные случаи. Обобщение их приводит к неоправданному повышению трудоемкости проверок. Антецедент, имеющий вид равенства, берется в посылках, прочие - обрабатываются проверочными операторами.

$\forall_{abc}(a = b\ \&\ 0 \le b + c \to 0 \le a + c)$

Выражения b, c константные, a - неконстантная сумма. Уровень срабатывания равен 3.

$\forall_{ab}(a = b\ \&\ a \le 0 \to b \le 0)$

$\forall_{ab}(a = b\ \&\ 0 \le a \to 0 \le b)$

Выражение b имеет заголовок "крд". Введен сильный ограничитель трудоемкости. Уровень срабатывания равен 3.

$\forall_{ab}(a = b\ \&\ 0 \le a \to 0 \le b)$

В отличие от предыдущего приема, здесь нет ограничений на заголовок b, но зато требуется наличие комментария "верхняяоценка". Введен средний ограничитель трудоемкости. Уровень срабатывания тот же.

$\forall_{abc}(a = b\ \&\ c \le b \to 0 \le a - c)$

$\forall_{ab}(a = -b\ \&\ 0 \le b \to a \le 0)$

$\forall_{abcd}(\text{длина}(a) = bc/d\ \&\ 0 < b\ \&\ 0 < d \to 0 \le c)$

Уровень срабатывания равен 3.

$\forall_{abcd}(0 < b\ \&\ 0 < d\ \&\ ab = cd \to 0 \le ac)$

Вырождение сомножителей в единицу не разрешаются. Общие части идентифицируются по пересечению групп сомножителей. Уровень срабатывания приема равен 4.

$\forall_{abc}(-a = bc\ \&\ 0 < c\ \&\ 0 \le a \to b \le 0)$

Выражение b имеет заголовок "крд". Введен сильный ограничитель трудоемкости. Уровень срабатывания равен 3.

$\forall_{abc}(a - b = c\ \&\ 0 \le b + c \to 0 \le a)$

Выражение a представляет собой сумму, содержащую неизвестные. Уровень срабатывания равен 4.

37. Знак произведения.

$\forall_{ab}(a \le 0 \ \& \ 0 \le b \to ab \le 0)$

$\forall_{ab}(0 \le a \ \& \ b \le 0 \to ab \le 0)$

$\forall_{ab}(a \le 0 \ \& \ b \le 0 \to 0 \le ab)$

$\forall_{ab}(0 \le a \ \& \ 0 \le b \to 0 \le ab)$

Первый антецедент берется в посылках, второй обрабатывается проверочным оператором. Переменная a идентифицируется с произведением. Уровень срабатывания равен 4.

38. Учет принадлежности промежутку.

$\forall_{abcde}(0 \le b \ \& \ a \in [b, c] \to 0 \le a)$

Переменные d, e суть "спрятанные" внутри сокращенного обозначения $[b, c]$ указатели принадлежности концов промежутка. Второй антецедент берется в посылках, первый обрабатывается проверочным оператором. Уровень срабатывания равен 1.

$\forall_{abc}(0 \le b \ \& \ a \in T \ \& \ T = [b, c] \to 0 \le a)$

$\forall_{abcT}(b \in T \ \& \ T = [b - a, c] \to 0 \le a)$

Два последних антецедента берутся в посылках. Так как приемы используются при работе с временными промежутками, указатели принадлежности концов равны 1 (промежутки замкнутые). Уровни срабатывания равны 3.

$\forall_{abcd}(a \in [b + c, d] \ \& \ 0 \le c \to b - a \le 0)$

$\forall_{abcd}(a \in [b + c, d] \ \& \ 0 \le c \to 0 \le a - b)$

$\forall_{abcd}(a \in [b, c + d] \ \& \ c \le 0 \to 0 \le d - a)$

Первый антецедент берется в посылках, второй обрабатывается проверочным оператором. Введен ускоряющий фильтр, проверяющий наличие посылки с вхождением символа "промежуток". Уровень срабатывания равен 3.

$\forall_{abT}(T = [a, b] \to 0 \le b - a)$

Антецедент берется в посылках. Текущая задача имеет понятия, относящиеся к разделу "физика" - по умолчанию предполагается, что в этом случае рассматриваемые промежутки невырожденные. Уровень срабатывания равен 3.

39. Элемент набора натуральных чисел.

$\forall_{abni}(i \in \{1, \ldots, n\} \ \& \ 0 \le b + 1 \ \& \ \text{кортеж}(a, n, \mathbf{N}) \to 0 \le a(i) + b)$

Третий антецедент берется в посылках, первые два - обрабатываются проверочными операторами. Переменная a - не функциональная. Уровень срабатывания равен 4.

40. Неравенство для арккосинуса.

$\forall_{abcdpq}(b < 0 \ \& \ 0 < pq \ \& \ 0 \le a + b\pi p/q \ \& \ c < p \arccos d/q \to 0 \le a + bc)$

Четвертый антецедент берется в посылках, причем допускается нестрогое неравенство. Остальные антецеденты обрабатываются проверочными операторами. Выражение c неконстантное. Уровень срабатывания равен 3.

41. Усмотрение знака логарифма.

    $\forall_{ab}(0 < a - 1 \ \& \ 0 \leq b - 1 \rightarrow 0 \leq \log_a b)$

    $\forall_{ab}(0 < a - 1 \ \& \ b - 1 \leq 0 \rightarrow log_a b \leq 0)$

    Антецеденты обрабатываются проверочными операторами. Указатель "спуск(1)" блокирует альтернативные попытки после проверки первого антецедента. Уровень срабатывания равен 2.

    $\forall_{a}(0 \leq a - e \rightarrow 0 \leq \ln\ln a)$

    Антецедент обрабатывается проверочным оператором. Уровень срабатывания равен 1.

42. Неотрицательность модуля.

    $\forall_{a}(0 \leq |a|)$

    Уровень срабатывания равен 1.

43. Сумма выражения и его модуля.

    $\forall_{a}(0 \leq -a + |a|)$

    $\forall_{a}(0 \leq a + |a|)$

    Уровень срабатывания равен 1.

44. Усмотрение знака суммы путем разложения на множители.

    $\forall_{ab}(b = a \ \& \ 0 \leq b \rightarrow 0 \leq a)$

    Выражение a представляет собой сумму двух слагаемых. Первый антецедент выделен указателем "идентификатор" и обращается к оператору "видумножение". Проверяется, что b не является суммой. Второй антецедент обрабатывается проверочным оператором. Прием блокируется, если a имеет дробное слагаемое либо содержит тригонометрическую операцию. Он блокируется также при наличии посылки вида "стремится(...)". Уровень срабатывания равен 3.

45. Неравенства с синусом.

    $\forall_{abc}(0 \leq b - 1 \ \& \ 0 \leq c - 1 \rightarrow 0 \leq c - (\sin a)^b)$

    $\forall_{abc}(0 \leq b - 1 \ \& \ 0 \leq c - 1 \rightarrow 0 \leq c + (\sin a)^b)$

    Выражения b, c константные. Антецеденты обрабатываются проверочными операторами. Уровень срабатывания равен 2.

    $\forall_{a}(0 \leq a \ \& \ 0 \leq \pi - a \rightarrow 0 \leq \sin a)$

    $\forall_{a}(0 \leq a - \pi \ \& \ 0 \leq 2\pi - a \rightarrow \sin a \leq 0)$

    Антецеденты обрабатываются проверочными операторами. Должна существовать посылка, либо содержащая символ "угол", либо представляющая собой неравенство без тригонометрических операций, имеющее общие параметры с a. Второй прием сопровождается сильным ограничителем трудоемкости. Уровень срабатывания равен 3.

    $\forall_{a}(0 \leq \sin a \ \& \ 0 \leq \cos a \rightarrow 0 \leq \sin(2a))$

    $\forall_{a}(\sin a \leq 0 \ \& \ \cos a \leq 0 \rightarrow 0 \leq \sin(2a))$

    Уровень срабатывания равен 2.

$\forall_a(0 \le \sin a \ \& \ 0 \le \cos a \to 0 \le \sin(a + \pi/4))$

Уровень срабатывания равен 1.

$\forall_{ABCab}(a = \angle(ABC) \ \& \ b = \angle(BAC) \to 0 \sin(a + b))$

$\forall_{ABCab}(a = \angle(ABC) \ \& \ b = \angle(BAC) \ \& \ 0 \le b - a \to \sin(a - b) \le 0)$

Антецеденты берутся в посылках. Уровень срабатывания равен 3.

$\forall_{abcdx}(0 \le a - x \ \& \ 0 \le x + b \ \& \ 0 < \pi - |(a + b)c| \ \& \ 0 \le \sin(-bc + d) \ \& \ 0 \le \sin(ac + d) \to 0 \le \sin(cx + d))$

$\forall_{abcdx}(0 \le a - x \ \& \ 0 \le x + b \ \& \ 0 < \pi - |(a + b)c| \ \& \ \sin(-bc + d) \le 0 \ \& \ \sin(ac + d) \le 0 \to \sin(cx + d) \le 0)$

Первые два антецедента берутся в посылках, причем допускаются строгие неравенства. Остальные антецеденты обрабатываются проверочными операторами. Выражения a, b, c, d константные, x - переменная. Уровень срабатывания приемов равен 3.

$\forall_{abcdkx}(0 \le a - x \ \& \ 0 \le x + b \ \& \ |(a + b)c| = \pi \ \& \ -bc + d = \pi k \to 0 \le \sin(cx + d))$

Первые два антецедента берутся в посылках, причем допускаются строгие неравенства. Остальные антецеденты выделены указателем "идентификатор" и используют нормализаторы общей стандартизации. Выражения a, b, c, d константные, x - переменная, k - целочисленная четная константа. Уровень срабатывания равен 3.

46. Неравенства с косинусом.

$\forall_a(0 \le a \ \& \ 0 \le \pi - 2a \to 0 \le \cos a)$

$\forall_a(0 \le 2a - \pi \ \& \ 0 \le \pi - a \to \cos a \le 0)$

Антецеденты обрабатываются проверочными операторами. Должна существовать посылка, либо содержащая символ "угол", либо представляющая собой неравенство без тригонометрических операций, имеющее общие параметры с a. Второй прием сопровождается сильным ограничителем трудоемкости. Уровень срабатывания первого приема равен 3, второго - 4.

$\forall_a(0 \le a + \pi/2 \ \& \ 0 \le \pi/2 - a \to 0 \le \cos a)$

$\forall_a(0 \le a - \pi/2 \ \& \ 0 \le 3\pi/2 - a \to \cos a \le 0)$

Аналогично предыдущему. Приемы имеют сильные ограничители трудоемкости. Уровни срабатывания равны, соответственно, 3 и 4.

$\forall_{ab}(0 \le b - 1 \to 0 \le 1 - |\cos a|^b)$

$\forall_{ab}(0 \le b - 1 \to 0 \le 1 + |\cos a|^b)$

Антецедент обрабатывается проверочным оператором. Выражение b константное. Уровень срабатывания равен 2.

$\forall_{abc}(0 \le b - 1 \ \& \ 0 \le c - 1 \to 0 \le c - (\cos a)^b)$

$\forall_{abc}(0 \le b - 1 \ \& \ 0 \le c - 1 \to 0 \le c + (\cos a)^b)$

Антецеденты обрабатываются проверочными операторами. Выражения b, c константные. Уровень срабатывания равен 2.

$\forall_{abc}(b \le a \ \& \ a \le c \ \& \ 0 \le b + \pi/2 \ \& \ 0 \le \pi/2 - c \to 0 \le \cos a)$

Первые два антецедента берутся в посылках, остальные обрабатываются проверочными операторами. Выражение a неконстантное. Введен сильный ограничитель трудоемкости. Уровень срабатывания равен 3.

$\forall_{abcdx}(0 \leq a - x \;\&\; 0 \leq x + b \;\&\; 0 < \pi - |(a + b)c| \;\&\; 0 \leq \cos(-bc + d) \;\&\; 0 \leq \cos(ac + d) \rightarrow 0 \leq \cos(cx + d))$

$\forall_{abcdx}(0 \leq a - x \;\&\; 0 \leq x + b \;\&\; 0 < \pi - |(a + b)c| \;\&\; \cos(-bc + d) \leq 0 \;\&\; \cos(ac + d) \leq 0 \rightarrow \cos(cx + d) \leq 0)$

$\forall_{abcdkx}(0 \leq a - x \;\&\; 0 \leq x + b \;\&\; |(a + b)c| = \pi \;\&\; -bc + d + \pi/2 = \pi k \rightarrow 0 \leq \cos(cx + d))$

Аналогично соответствующим приемам для неравенств с синусом.

47. Неравенства с тангенсом.

    $\forall_a(0 \leq a \;\&\; 0 \leq \pi - 2a \rightarrow 0 \leq \operatorname{tg} a)$

    Антецеденты обрабатываются проверочными операторами. Должна существовать посылка, либо содержащая символ "угол", либо представляющая собой неравенство без тригонометрических операций, имеющее общие параметры с a. Уровень срабатывания равен 3.

    $\forall_a(0 \leq \sin a \;\&\; 0 \leq \cos a \rightarrow 0 \leq \operatorname{tg} a)$

    Созданы две версии приема. В одной из них первый антецедент берется в посылках, а второй обрабатывается проверочным оператором, в другом - второй антецедент берется в посылках, а первый обрабатывается проверочным оператором. Допускается использование посылки, являющейся строгим неравенством. Уровень срабатывания равен 2.

    $\forall_a(0 < \operatorname{ctg} a \rightarrow 0 \leq \operatorname{tg} a)$

    Антецедент берется в посылках. Уровень срабатывания равен 2.

48. Неравенства с котангенсом.

    $\forall_a(0 \leq a \;\&\; 0 \leq \pi - 2a \rightarrow 0 \leq \operatorname{ctg} a)$

    $\forall_a(0 < \operatorname{tg} a \rightarrow 0 \leq \operatorname{ctg} a)$

    Аналогично случаю тангенса.

49. Суммы с тригонометрическими функциями.

    (a) Трехчлен из теоремы косинусов.
    $\forall_{abc}(0 \leq a^2 + b^2 - 2ab\cos c)$
    $\forall_{abc}(0 \leq a^2 + b^2 + 2ab\cos c)$
    Уровень срабатывания равен 3.

    (b) Сумма единицы и произведения положительных степеней синусов и косинусов.
    $\forall_a(0 \leq 1 + a)$
    Фильтр "не(контекст(вид(x1 умножение(x2 x3))единица(1 x3)заменазнака (минус x3)не(контекст(вид(x2 степень(x4 x5)) символ(x4 синус косинус) не(пересекаются (параметры(x5) параметры(x4)))легковидеть(меньше(0 x5)) единица(1 x5)))))" обеспечивает проверку отсутствия множителей выражения a, не являющихся положительными степенями синусов и косинусов. Дополнительно требуется, чтобы основания и показатели степеней не

имели общих параметров. Это требование введено лишь для отсечения случаев, которые могли бы замедлить работу проверочного оператора. Уровень срабатывания равен 3.

(c) Линейные комбинации с косинусом и синусом.

$\forall_{abc}(0 \leq a \ \& \ 0 \leq b \rightarrow 0 \leq a\cos c - b\cos c + a + b)$

$\forall_{abc}(0 \leq a \ \& \ 0 \leq b \rightarrow 0 \leq (a-b)\cos c + a + b)$

Антецеденты обрабатываются проверочными операторами. Вместо косинусов допускаются синусы. Уровень срабатывания равен 3.

(d) Сумма константы и произведения синуса либо косинуса на число.

$\forall_{abx}(0 \leq a - |b| \ \& \ 0 < a \rightarrow 0 \leq a + b\cos x)$

$\forall_{abx}(0 \leq a - |b| \ \& \ 0 < a \rightarrow 0 \leq a + b\sin x)$

Антецеденты обрабатываются проверочными операторами. Выражения a, b константные. Уровень срабатывания равен 3.

50. Неравенства с арккосинусом.

$\forall_a(0 \leq \arccos a)$

Уровень срабатывания равен 1.

$\forall_{ab}(0 \leq a - \pi \rightarrow \arccos b \leq a)$

Антецедент обрабатывается проверочным оператором. Уровень срабатывания равен 2.

$\forall_{ab}(0 \leq a \rightarrow 0 \leq \pi - \arccos b + a)$

$\forall_{ab}(a \leq 0 \rightarrow \arccos b - \pi + a \leq 0)$

$\forall_{ab}(0 \leq a + \pi/2 \ \& \ 0 \leq b \rightarrow 0 \leq \pi - \arccos b + a)$

Антецеденты обрабатываются проверочными операторами. Уровень срабатывания равен 3.

51. Неравенства с арксинусом.

$\forall_a(0 \leq \pi/2 - \arcsin a)$

Уровень срабатывания равен 1.

$\forall_a(0 \leq a \rightarrow 0 \leq \arcsin a)$

$\forall_a(a \leq 0 \rightarrow \arcsin a \leq 0)$

$\forall_{ab}(0 \leq 2a - \pi \rightarrow \arcsin b \leq a)$

Антецеденты обрабатываются проверочными операторами. Уровень срабатывания равен 2.

$\forall_a(0 \leq \pi - \arcsin a)$

$\forall_a(0 \leq \pi + \arcsin a)$

Уровень срабатывания равен 3.

52. Неравенства с арктангенсом.

$\forall_a(a \leq 0 \rightarrow \operatorname{arctg} a \leq 0)$

$\forall_a(0 \leq a \rightarrow 0 \leq \operatorname{arctg} a)$

Антецеденты обрабатываются проверочным оператором. Уровень срабатывания равен 2.

$\forall_{ab}(0 < b \ \& \ 0 \leq 1 - b \rightarrow (\operatorname{arctg} a)^b \leq \pi/2)$

$\forall_{a}(-\pi/2 \leq \operatorname{arctg} a)$

$\forall_{a}(0 \leq \pi + 2\operatorname{arctg} a)$

$\forall_{a}(0 \leq \pi - 2\operatorname{arctg} a)$

Антецедент у первого приема обрабатывается проверочным оператором. Уровень срабатывания равен 1.

$\forall_{abc}(0 < c \ \& \ 0 \leq 1 - c \ \& \ 0 \leq 2b - \pi \rightarrow 0 \leq (\operatorname{arctg} a)^c + b)$

$\forall_{abc}(0 < c \ \& \ 0 \leq 1 - c \ \& \ 0 \leq 2b - \pi \rightarrow 0 \leq -(\operatorname{arctg} a)^c + b)$

$\forall_{ab}(2b + \pi \leq 0 \rightarrow \operatorname{arctg} a + b \leq 0)$

$\forall_{ab}(2b + \pi \leq 0 \rightarrow b \leq \operatorname{arctg} a)$

$\forall_{ab}(0 \leq 2b - \pi \rightarrow \operatorname{arctg} a \leq b)$

Антецеденты обрабатываются проверочными операторами. Уровень срабатывания равен 2.

53. Неравенства для геометрических величин.

    (a) Неотрицательность расстояния.

    $\forall_{AB}(0 \leq l(AB))$

    $\forall_{ab}(0 \leq \text{расстдопрямой}(a, b))$

    $\forall_{ab}(0 \leq \text{расстмежду}(a, b))$

    Уровень срабатывания равен 1.

    $\forall_{acAB}(ac = l(AB) \ \& \ 0 < a \rightarrow 0 \leq c)$

    $\forall_{acAB}(ac = \text{расстдопрямой}(A, B) \ \& \ 0 < a \rightarrow 0 \leq c)$

    $\forall_{acAB}(ac = \text{расстмежду}(A, B) \ \& \ 0 < a \rightarrow 0 \leq c)$

    Первый антецедент берется в посылках, второй - обрабатывается проверочным оператором. Выражение a константное. Уровень срабатывания равен 3. Напомним, что $l(AB)$ - расстояние между точками A, B, "расстдопрямой (a, b)" - расстояние от точки a до прямой b, "расстмежду(a, b)" - расстояние между двумя множествами точек (точная нижняя грань расстояний между точками этих множеств).

    (b) Неотрицательность угла.

    $\forall_{ABC}(0 \leq \angle(ABC))$

    $\forall_{AB}(0 \leq \text{уголмежду}(A, B))$

    $\forall_{ABCd}(0 \leq \text{двугругол}(A, B, C, d))$

    $\forall_{ABC}(0 \leq \text{Уголмежду}(A, B, C))$

    Уровень срабатывания равен 1.

    $\forall_{ABCd}(d = \angle(ABC) \rightarrow 0 \leq d)$

    $\forall_{ABd}(d = \text{уголмежду}(A, B) \rightarrow 0 \leq d)$

    $\forall_{ABCDd}(d = \text{двугругол}(A, B, C, D) \rightarrow 0 \leq d)$

    Антецедент берется в посылках. Уровень срабатывания равен 3. Напомним обозначения "уголмежду", "двугругол", "Уголмежду". "уголмежду(A, B)"

- величина угла между прямыми либо плоскостями либо векторами A, B. Для прямых и плоскостей она выбирается между 0 и $\pi/2$. "двугругол(A, B, C, D)" - величина двугранного угла, определяемого плоскостями A, B, точкой C вне этих плоскостей и указателем D области, к которой относится точка по отношению к рассматриваемому углу (4 варианта). "Уголмежду (A, B, C)" - величина того угла между прямыми A, B, в котором лежит прямая C, относящаяся к той же плоскости.

(c) Величина угла не превосходит пи.

$\forall_{ABCmnpq}(0 \leq pn - qm \rightarrow 0 \leq p\pi/q - m\angle(ABC)/n)$

Переменные m, n, p, q идентифицируются с натуральными константами. Антецедент выделен указателем "программа". Уровень срабатывания равен 1.

$\forall_{abABC}(a = \angle(ABC)\ \&\ 0 \leq b - \pi \rightarrow 0 \leq b - a)$

Первый антецедент берется в посылках, второй - обрабатывается проверочным оператором. Фильтр "контекст(посылка(x3)входит(угол x3))" нужен для ускоренного отсечения попыток применить прием в негеометрических задачах. Уровень срабатывания равен 3.

$\forall_{ABCa}(0 \leq a \rightarrow 0 \leq a - \angle(ABC) + \pi)$

Антецедент обрабатывается проверочным оператором. Введен очень сильный ограничитель трудоемкости. Уровень срабатывания равен 4.

$\forall_{ABmn}(0 \leq n - m \rightarrow 0 \leq \pi - m\text{уголмежду}(A, B)/n)$

Переменные m, n идентифицируются с натуральными константами. Антецедент обрабатывается проверочным оператором. Уровень срабатывания равен 1.

$\forall_{ABab}(a = \text{уголмежду}(A, B)\ \&\ 0 \leq b - \pi \rightarrow 0 \leq b - a)$

Первый антецедент берется в посылках, второй - обрабатывается проверочным оператором. Уровень срабатывания равен 3.

$\forall_{ABCDmn}(0 \leq n - m \rightarrow 0 \leq \pi - m\text{двугругол}(A, B, C, D)/n)$

$\forall_{ABab}(a = \text{двугругол}(A, B)\ \&\ 0 \leq b - \pi \rightarrow 0 \leq b - a)$

$\forall_{ABCDmn}(0 \leq n - m \rightarrow 0 \leq \pi - m\text{Уголмежду}(A, B, C, D)/n)$

$\forall_{ABab}(a = \text{Уголмежду}(A, B)\ \&\ 0 \leq b - \pi \rightarrow 0 \leq b - a)$

Приемы аналогичны рассмотренным выше.

(d) Усмотрение острого либо тупого угла.

i. Углы трапеции.

$\forall_{ABCD}(\text{трапеция}(ABCD) \rightarrow 0 \leq \pi/2 - \angle(BAD))$
$\forall_{ABCD}(\text{трапеция}(ABCD) \rightarrow 0 \leq \angle(ABC) - \pi/2)$
$\forall_{aABCD}(\text{трапеция}(ABCD)\ \&\ 0 \leq a - \pi/2 \rightarrow 0 \leq a - \angle(CAD))$
$\forall_{aABCD}(\text{трапеция}(ABCD)\ \&\ 0 \leq a - \pi/2 \rightarrow 0 \leq a - \angle(BCA))$
Указатель "контрсерия(фикс(1))" обеспечивает возможность изменения порядка перечисления вершин трапеции на противоположный. Уровень срабатывания равен 2.

ii. Вписанные углы.

$\forall_{ABCD}(C \in \text{окружность}(AB)\ \&\ D \in \text{окружность}(AB)\ \&\ \text{однасторона}(A, C, \text{прямая}(BD)) \rightarrow 0 \leq \pi/2 - \angle(BCD))$

Напомним, что "однасторона(A, C, прямая(CD))" означает расположение точек по одну сторону от прямой CD. Таким образом, речь идет о вписанном угле, вершина которого расположена от хорды по ту же сторону, что и центр окружности. Первый и второй антецеденты обрабатываются идентифицирующими операторами, третий антецедент - проверочным оператором. Уровень срабатывания приема равен 4.

$\forall_{ABCD}(C \in \text{окружность}(AB)\ \&\ D \in \text{окружность}(AB)\ \&\ \text{разныестороны}(A, C, \text{прямая}(BD)) \to 0 \leq \angle(CBD) - \pi/2)$

Уровень срабатывания равен 2.

$\forall_{ABCDE}(C \in \text{окружность}(AB)\ \&\ D \in \text{окружность}(AB)\ \&\ E \in \text{окружность}(AB)\ \&\ \text{разныестороны}(A, C, \text{прямая}(DE)) \to 0 \leq \pi/2 - \angle(CDE))$

Первые три антецедента обрабатываются идентифицирующими операторами, четвертый - проверочным оператором. Введен сильный ограничитель трудоемкости. Уровень срабатывания равен 3.

$\forall_{ABCD}(B \in \text{окружность}(AD)\ \&\ C \in \text{окружность}(AD) \to 0 \leq \pi/2 - \angle(CBA))$

Рассматривается вписанный угол, одна сторона которого является диаметром. Антецеденты обрабатываются идентифицирующими операторами. Уровень срабатывания равен 1.

iii. Усмотрение прямоугольного треугольника, имеющего данный угол.

$\forall_{ABC}(\text{прямая}(AB) \perp \text{прямая}(BC) \to 0 \leq \pi/2 - \angle(BAC))$

Антецедент обрабатывается идентифицирующим оператором. Заметим, что компилятор автоматически обобщает теорему приема: на луче AB ищется точка B', через которую проходит перпендикуляр к AB, пересекающийся с лучом AC в некоторой точке C'. Уровень срабатывания равен 3.

$\forall_{ABCa}(a = \angle(BAC)\ \&\ \text{прямая} \perp \text{прямая}(BC) \to 0\pi - 2a)$

Первый антецедент берется в посылках, второй обрабатывается идентифицирующим оператором. Уровень срабатывания равен 4.

iv. Угол между стороной и диагональю параллелограмма.

$\forall_{ABCD}(\text{актив}(\angle(BAD))\ \&\ \text{прямая}(BC) \parallel \text{прямая}(AD)\ \&\ \text{прямая}(AB) \parallel \text{прямая}(CD)\ \&\ 0 \leq \pi/2 - \angle(BAD) \to 0 \leq \pi/2 - \angle(BAC))$

Первые три антецедента обрабатываются идентифицирующими операторами. Четвертый антецедент, указывающий, что диагональ проведена из острого угла параллелограмма, обрабатывается проверочным оператором. Введен сильный ограничитель трудоемкости. Уровень срабатывания равен 3.

v. Переход к вспомогательному углу с помощью равенства из посылок.

$\forall_{ABCa}(\angle(ABC) = a\ \&\ 0 \leq \pi/2 - \angle(ABC) \to 0 \leq \pi/2 - a)$

$\forall_{ABCa}(\angle(ABC) = a\ \&\ 0 \leq \pi/2 - \angle(ABC) \to 0 \leq \pi - 2a)$

Первый антецедент берется в посылках, второй - обрабатывается проверочным оператором. Ускоряющий фильтр проверяет наличие посылки, содержащей символ "угол". Уровень срабатывания равен 3.

vi. Угол между боковой стороной трапеции и отрезком, проведенным к точке на противоположной боковой стороне.

$\forall_{ABCDE}(\text{трапеция}(ABCD)\ \&\ E \in \text{отрезок}(CD) \to 0 \leq \pi/2 - \angle(BAE))$

Первый антецедент берется в посылках, второй - обрабатывается идентифицирующим оператором. Уровень срабатывания равен 3.

vii. Центральный угол, опирающийся на конец хорды, параллельной другой его стороне.

$\forall_{ABCDE}(C \in$ окружность(AB) & $D \in$ окружность(AB) & $E \in$ отрезок(AD) & прямая$(EC) \parallel$ прямая(AB) & однасторона$(B, C,$ прямая$(AD)) \to 0 \leq \pi/2 - \angle(BAC))$

Первые антецеденты обрабатываются идентифицирующими операторами, последний - проверочным оператором. Уровень срабатывания равен 4.

viii. Попытка перехода к неравенству с половиной пи.

$\forall_a(0 \leq \pi/2 - a \to 0 \leq \pi - 2a)$

Антецедент берется в посылках, причем допускается строгое неравенство. Уровень срабатывания равен 1.

$\forall_{ABC}(0 \leq \pi/2 - \angle(ABC) \to 0 \leq \pi - 2\angle(ABC))$

Антецедент обрабатывается проверочным оператором. Уровень срабатывания равен 3.

ix. Переход к дополнительному углу.

$\forall_{ABCDEF}(\angle(ABC) + \angle(DEF) = \pi$ & $0 < \pi/2 - \angle(DEF) \to 0 \leq \angle(ABC) - \pi/2)$

Оба антецедента берутся в посылках, причем допускается случай нестрогого неравенства. Уровень срабатывания равен 4.

x. Угол между секущей и радиусом.

$\forall_{ABCDE}(D \in$ окружность(AB) & $E \in$ окружность(AB) & $D \in$ отрезок(CE) & разныеточки$(D, E) \to 0 \leq \angle(CDA) - \pi/2)$

Первые три антецедента обрабатываются идентифицирующими операторами, четвертый - проверочным оператором "разныеточки", устанавливающим различие точек. Уровень срабатывания равен 3.

$\forall_{ABCD}(C \in$ окружность(AB) & $D \in$ окружность$(AB) \to 0 \leq \pi/2 - \angle(CDA))$

Антецеденты обрабатываются идентифицирующими операторами. Уровень срабатывания равен 3.

xi. Угол при основании прямоугольной трапеции.

$\forall_{ABCD}($прямая$(BC) \parallel$ прямая(AD) & прямая$(CD) \perp$ прямая(AD) & $0 \leq l(AD) - l(BC) \to 0 \leq \pi/2 - \angle(BAD))$

Первые два антецедента обрабатываются идентифицирующими операторами, третий - проверочным оператором. Введен сильный ограничитель трудоемкости. Уровень срабатывания равен 3.

xii. Переход к рассмотрению противоположного угла ромба либо параллелограмма.

$\forall_{ABCD}($ромб$(ABCD)$ & актив$(\angle(ADC))$ & $0 \leq \angle(ADC) - \pi/2 \to 0 \leq \angle(ABC) - \pi/2)$

Первый антецедент берется в посылках, причем вместо символа "ромб" допускается символ "параллелограмм". Разрешается также циклическая перестановка операндов A, B, C, D. Второй антецедент обрабатывается идентифицирующим оператором, третий - проверочным оператором. Введен сильный ограничитель трудоемкости. Уровень срабатывания равен 4.

xiii. Усмотрение прямого подугла.

$\forall_{ABCD}($прямая$(AB) \perp$ прямая(BD) & $D \in$ отрезок$(AC) \to 0 \leq \angle(ABC) - \pi/2)$

Антецеденты обрабатываются идентифицирующими операторами. Уровень срабатывания равен 3.

xiv. Внутренняя точка четырехугольника.
$\forall_{ABCDE}(E \in \text{фигура}(ABCD)\ \&\ \text{квадрат}(ABCD) \rightarrow 0 \leq \pi/2 - \angle(DAE))$
$\forall_{ABCDE}(E \in \text{фигура}(ABCD)\ \&\ \text{квадрат}(ABCD) \rightarrow 0 \leq \pi/2 - \angle(BAE))$
Антецеденты берутся в посылках, причем вместо символа "квадрат" допускается символ "прямоугольник". Допускаются циклические перестановки точек. Уровень срабатывания равен 3.
$\forall_{ABCDE}(E \in \text{фигура}(ABCD)\ \&\ \text{четырехугольник}(ABCD)\ \&\ 0 \leq \pi/2 - \angle(BAD) \rightarrow 0 \leq \pi/2 - \angle(DAE))$
Первые два антецедента берутся в посылках, третий - обрабатывается проверочным оператором. Вместо символа "четырехугольник" допускаются символы "ромб", "трапеция", "параллелограмм". Допускаются произвольные циклические перестановки точек, а также изменение их порядка на противоположный. Введен сильный органичитель трудоемкости. Уровень срабатывания равен 4.

xv. Угол при основании равнобедренного треугольника.
$\forall_{ABC}(l(AB) = l(BC) \rightarrow 0 \leq \pi/2 - \angle(BAC))$
Антецедент выделен указателем "идентификатор". Обе части равенства обрабатываются нормализатором "нормрасстояние", учитывающим наличие в контексте равенств, явно задающих величину расстояния. Уровень обращения равен 4.

xvi. Заданная доля величины угла.
$\forall_{ABCamn}(0 \leq n - 2m\ \&\ a = \angle(ABC) \rightarrow 0 \leq \pi/2 - am/n)$
Второй антецедент берется в посылках. Выражения m, n - десятичные константы, a - переменная. Первый антецедент выделен указателем "программа". Уровень срабатывания равен 4.

xvii. Угол между вектором и плоскостью.
$\forall_{ABCDE}(0 \leq \pi/2 - \text{уголмежду}(\text{вектор}(AB), \text{плоскость}(CDE)))$
Уровень обращения равен 1.

xviii. Угол между двумя прямыми либо двумя плоскостями.
$\forall_{ABCD}(0 \leq \pi/2 - \text{уголмежду}(\text{прямая}(AB), \text{прямая}(CD)))$
$\forall_{ABCDEF}(0 \leq \pi/2 - \text{уголмежду}(\text{плоскость}(ABC), \text{плоскость}(DEF)))$
$\forall_{ABCDE}(0 \leq \pi/2 - \text{уголмежду}(\text{плоскость}(ABC), \text{прямая}(DE)))$
Уровень срабатывания равен 1.
$\forall_{aABCDE}(a = \text{уголмежду}(\text{плоскость}(ABC), \text{прямая}(DE)) \rightarrow 0 \leq \pi/2 - a)$
$\forall_{aABCDEF}(a = \text{уголмежду}(\text{плоскость}(ABC), \text{плоскость}(DEF)) \rightarrow 0 \leq \pi/2 - a)$
$\forall_{aABCD}(a = \text{уголмежду}(\text{прямая}(AB), \text{прямая}(CD)) \rightarrow 0 \leq \pi/2 - a)$
Антецедент берется в посылках. Уровень срабатывания равен 3.

xix. Использование посылок.
$\forall_{ABCa}(a = \angle(ABC)\ \&\ \angle(ABC) < \pi/2 \rightarrow 0 \leq \pi/2 - a)$
Антецеденты берутся в посылках. Уровень срабатывания равен 4.
$\forall_{ABCa}(\angle(ABC) + \angle(CAB) < a\ \&\ 0 \leq \pi/2 - a \rightarrow 0 \leq \angle(ACB) - \pi/2)$
$\forall_{ABCpq}(p\angle(ABC) = q\angle(ACB)\ \&\ 0 \leq p - q \rightarrow 0 \leq \pi/2 - \angle(ABC))$
Первый антецедент берется в посылках, второй обрабатывается проверочным оператором. Уровень срабатывания равен 4.

xx. Точка внутри прямого угла.

$\forall_{ABCD}$(прямая$(AB) \perp$ прямая(BC) & однасторона$(A, D,$ прямая$(BC))$ & однасторона$(C, D,$ прямая$(AB)) \to 0 \leq \pi/2 - \angle(ABD))$

Первый антецедент обрабатывается идентифицирующим оператором, второй и третий - проверочными операторами. Введен сильный ограничитель трудоемкости. Уровень срабатывания равен 4.

(e) Неотрицательность коэффициента пропорциональности расстояний.

$\forall_{ABCDa}(al(AB) = l(CD)$ & разныеточки$(A, B) \to 0 \leq a)$

$\forall_{ABCa}(al(AB) =$ длина(C) & разныеточки$(A, B) \to 0 \leq a)$

Первый антецедент берется в посылках, второй - обрабатывается проверочным оператором. Уровень срабатывания равен 3.

(f) Неотрицательность площади.

$\forall_a(0 \leq S(a))$

Уровень срабатывания равен 1.

$\forall_{ab}(a = S(b) \to 0 \leq a)$

Антецедент берется в посылках. Уровень срабатывания равен 2.

(g) Неотрицательность периметра.

$\forall_a(0 \leq$ периметр$(a))$

Уровень срабатывания равен 1.

(h) Соотношение длин сторон равнобедренного треугольника, у которого известен угол при основании.

$\forall_{ABCabc}(l(AB) = l(BC)$ & $\angle(BAC) = a$ & разныеточки(A, C) & $0 \leq b + 2c\cos a \to 0 \leq bl(AB) + cl(AC))$

Проверяется неотрицательность линейной комбинации длин боковой стороны и основания. Первые два антецедента выделены указателем "идентификатор": после усмотрения равенства расстояний AB, BC определяется величина a угла при основании. Проверяется, что выражения a, b, c - константные. Третий и четвертый антецеденты обрабатываются проверочными операторами: убеждаемся, что точки A, C различны, выражаем отношение длин сторон через косинус a и проверяем полученное неравенство. Уровень срабатывания равен 4.

(i) Против большего угла треугольника лежит большая сторона.

$\forall_{ABC}$(актив$(\angle(BAC))$ & актив$(\angle(BCA))$ & $0 \leq \angle(BAC) - \angle(BCA)$ & разныепрямые(прямая(AB), прямая$(AC)) \to 0 \leq l(BC) - l(AB))$

Первые два антецедента обрабатываются идентифицирующими операторами. Посылки вида "актив(угол(...))", "актив(расстояние(...))" и т.п. используются для выделения рассматриваемых в задаче параметров чертежа. Их наличие удобно для инициализации попыток применения приемов. В данном случае обработка антецедентов "актив" позволяет убедиться в том, что углы BAC, BCA уже рассматриваются в задаче. В частности, выделены прямые AB, AC, BC, причем точки A, B, C попарно различны. Третий и четвертый антецеденты обрабатываются проверочными операторами. Уровень срабатывания равен 4.

$\forall_{ABC}$(актив$(\angle(BAC))$ & $0 \leq \angle(BAC) - \pi/2$ &
разныепрямые(прямая(AB), прямая$(AC)) \to 0 \leq l(BC) - l(AB))$

Первый антецедент обрабатывается идентифицирующим оператором, второй и третий - проверочными операторами. Уровень срабатывания равен 4.

(j) Использование равенства для суммы расстояний.
$\forall_{ABCDa}(l(AB)+l(CD)=a \to 0 \le a-l(AB))$
Антецедент берется в посылках. Уровень срабатывания равен 3.

(k) Сравнение длин хорд окружности.
$\forall_{ABCDEF}(C \in$ окружность(AB) & $D \in$ окружность(AB) & $F \in$ дуга(ACD) & $E \in$ окружность(AB) & $0 \le l(CE)-l(CD) \to 0 \le l(EF)-l(DF))$

Из одной точки окружности проведены две хорды CD, CE, причем вторая хорда длиннее первой. Внутри меньшей дуги, отсекаемой первой хордой, на окружности взята точка F. Прием усматривает, что расстояние от нее до D не превосходит расстояния до E. Третий антецедент берется в посылках, принадлежность точек окружности усматривается с помощью идентифицирующих операторов, последний антецедент обрабатывается проверочным оператором. Уровень срабатывания равен 4.

(l) Разность квадратов радиусов двух пересекающихся окружностей и квадрат расстояния между их центрами.

$\forall_{ABCDabc}(a=l(AB)$ & $b=l(BD)$ & $c=l(AD)$ & разныестороны$(A, D,$прямая$(BC))$ & $C \in$ окружность(AB) & $C \in$ окружность(DB) & разныеточки$(B, C) \to 0 \le c^2+b^2-a^2)$

Первые три антецедента берутся в посылках. Пятый и шестой - обрабатываются идентифицирующими операторами, четвертый и седьмой - проверочными операторами. Уровень срабатывания равен 4.

(m) Линейная комбинация величин двух углов треугольника с рациональными коэффициентами, не превосходящими единицы.

$\forall_{abcdpqABC}(a=\angle(ABC)$ & $b=\angle(ACB)$ & $0 \le d-c$ & $0 \le q-p \to 0 \le \pi - ac/d - pb/q)$

Выражения c, d, p, q идентифицируются с натуральными константами. Первые два антецедента берутся в посылках, последние два - обрабатываются проверочными операторами. Уровень срабатывания равен 3.

(n) Неотрицательность площади поверхности и объема.
$\forall_a(0 \le$ площповерхности$(a))$
$\forall_a(0 \le$ объем$(a))$
Уровень срабатывания равен 1.

(o) Сумма квадратов длин сторон треугольника, примыкающих к острому либо прямому углу, не меньше квадрата длины третьей стороны.

$\forall_{ABCabc}(a=l(BC)$ & $b=l(AC)$ & $c=l(AB)$ & $0 \le \pi/2-\angle(BAC) \to 0 \le b^2+c^2-a^2)$

Первые три антецедента берутся в посылках, последний - обрабатывается проверочным оператором. Прием используется только при обращении из завершающего редактирования ответа геометрической задачи. Уровень срабатывания равен 4.

(p) Неотрицательность радиуса.
$\forall_a(0 \le$ радиус$(a))$

Уровень срабатывания равен 1.

(q) Неравенство треугольника.

$\forall_{ABCa}(a = l(BC) \rightarrow 0 \leq a - l(AB) + l(AC))$

Антецедент выделен указателем "идентификатор". Он проверяет равенство выражения a величине расстояния $l(BC)$, определяемой с помощью нормализатора "нормрасстояние". Уровень срабатывания равен 3.

(r) Неотрицательность длины.

$\forall_a(0 \leq \text{длина}(a))$

Уровень срабатывания равен 1.

$\forall_{ab}(a = \text{длина}(b) \rightarrow 0 \leq a)$

Антецедент берется в посылках. Используется ускоряющий фильтр, проверяющий наличие посылки с символом "длина". Уровень срабатывания равен 3.

(s) Разное.

$\forall_A(0 \leq \text{эксцентриситет}(A))$

Уровень срабатывания равен 1.

$\forall_{ab}(a = \text{эксцентриситет}(b) \rightarrow 0 \leq a)$

Уровень срабатывания равен 3.

$\forall_{ab}(0 \leq \sin(\text{уголмежду}(a, b)))$

Используется указатель "смравно(фикс(0 2 1))", позволяющий идентифицировать выражение $\sin(\text{уголмежду}(a, b))$ с термом $\sin A$, если в контексте содержится равенство $A = \text{уголмежду}(a, b)$. Уровень срабатывания приема равен 3.

$\forall_{abcd}(b < a \;\&\; 0 < c \;\&\; 0 \leq d + bc \rightarrow 0 \leq d + ac)$

$\forall_{abcd}(a < b \;\&\; 0 < c \;\&\; 0 \leq d - bc \rightarrow 0 \leq d - ac)$

Выражение a имеет своим заголовком один из символов "угол", "расстояние", "площадь". Первый антецедент берется в посылках, причем допускается нестрогое неравенство. Второй и третий антецеденты обрабатываются проверочными операторами. Уровень срабатывания равен 3.

$\forall_{ABCD}(\text{биссектриса}(ABCD) \rightarrow 0 \leq \pi - 2\angle(ABD))$

Уровень срабатывания равен 3.

$\forall_{ABpqmn}(0 < m \;\&\; 0 < p \;\&\; 0 < ml(AB) + n \;\&\; 0 \leq pn - mq \rightarrow 0 \leq pl(AB) + q)$

Третий антецедент берется в посылках, остальные обрабатываются проверочными операторами. Выражения n, q не содержат символов "угол", "расстояние". Уровень срабатывания равен 3.

54. Конечные сумма и произведение.

$$\forall_{fg}(0 \leq f(x) \rightarrow 0 \leq \prod_{x,g(x)} f(x)$$

$$\forall_{fg}(0 \leq f(x) \rightarrow 0 \leq \sum_{x,g(x)} f(x)$$

Переменные f, g - функциональные. Антецедент обрабатывается проверочным оператором, причем ему передается дополнительная посылка $g(x)$. Уровень срабатывания равен 1.

55. Учет предельных условий.

    $\forall_{abx}((x \to a\backslash b)\ \&\ 0 < a \to 0 \leq x)$

    Первый антецедент берется в посылках. Он представляет собой указатель на стремление переменной x к точке a, причем b - тип рассматриваемой окрестности этой точки (двусторонняя либо левая). Второй антецедент обрабатывается проверочным оператором. Уровень срабатывания равен 2. В случае правой окрестности строгое неравенство заменяется нестрогим:

    $\forall_{abx}((x \to a + 0)\ \&\ 0 \leq a \to 0 \leq x)$

    Для усмотрения нестрогого знака линейного относительно x выражения созданы следующие приемы:

    $\forall_{abcx}((x \to c + 0)\ \&\ 0 < a\ \&\ 0 \leq ac + b \to 0 \leq ax + b)$

    $\forall_{abcx}((x \to c - 0)\ \&\ 0 < a\ \&\ ac + b \leq 0 \to ax + b \leq 0)$

    $\forall_{abcx}((x \to c - 0)\ \&\ a < 0\ \&\ 0 \leq ac + b \to 0 \leq ax + b)$

    Первый антецедент берется в посылках, второй и третий - обрабатываются проверочными операторами. Выражения a, b не содержат переменной x. Уровень срабатывания первого приема равен 3, остальных приемов - 2. Для более сложных выражений с x вычисляется их предел в точке a:

    $\forall_{abcx}((x \to a\backslash b)\ \&\ c = \lim_{x \to a\backslash b} f(x)\ \&\ (c = \infty\ \vee\ 0 < c) \to 0 \leq f(x))$

    $\forall_{abcx}((x \to a\backslash b)\ \&\ c = \lim_{x \to a\backslash b} f(x)\ \&\ (c = -\infty\ \vee\ c < 0) \to f(x) \leq 0)$

    Первый антецедент берется в посылках, второй выделен указателем "идентификатор" и обращается к нормализатору "нормпредел", вычисляющему необходимый предел. Предварительно проверяется, что $f(x)$ содержит x. После обращения к нормализатору проверяется, что выражение c не содержит символа "предел". Переменная f - функциональная. Третий антецедент обрабатывается проверочным оператором. Уровень срабатывания равен 3. Если предел оказался равен 0, используется обращение к проверочным операторам, усматривающим локальную монотонность функции $f(x)$:

    $\forall_{ax}((x \to a{+}0)\ \&\ \lim_{x \to a+0} f(x) = 0\ \&\ \text{возрастаетвточке}(\lambda_x(f(x), x{-}\text{число}), a) \to 0 \leq f(x))$

    $\forall_{ax}((x \to a - 0)\ \&\ \lim_{x \to a-0} f(x) = 0\ \&\ \text{убываетвточке}(\lambda_x(f(x), x - \text{число}), a) \to 0 \leq f(x))$

    Антецеденты обрабатываются так же, как и выше. Введен сильный ограничитель трудоемкости. Уровень срабатывания равен 3. Для усмотрения неотрицательности предела числовой последовательности создан следующий прием:

    $\forall_{afi}(\lim(f) = a\ \&\ 0 \leq f(i) \to 0 \leq a)$

    Первый антецедент берется в посылках, причем посредством lim формульный редактор прорисовывает логический символ "пределпослед". Второй антецедент обрабатывается проверочным оператором, которому передается дополнительная посылка "натуральное(i)". Переменная f - обычная. Так как вспомогательная переменная i непосредственно не идентифицируется, для нее введен указатель "новаяпеременная(x9)". Уровень срабатывания равен 3.

56. Факториал.

    $\forall_n(0 \leq n!)$

Уровень срабатывания равен 1.

57. Число сочетаний.

    $\forall_{mn}(0 \leq C_m^n)$

    Уровень срабатывания равен 1.

58. Гиперболические функции.

    $\forall_a(0 \leq \operatorname{ch} a)$

    $\forall_a(0 \leq a \to 0 \leq \operatorname{sh} a)$

    $\forall_a(a \leq 0 \to \operatorname{sh} a \leq 0)$

    $\forall_a(0 \leq a \to 0 \leq \operatorname{th} a)$

    $\forall_a(0 \leq a \to 0 \leq \operatorname{cth} a)$

    Антецедент обрабатывается проверочным оператором. Уровень срабатывания равен 1.

    $\forall_{abc}(0 < a \;\&\; 0 \leq a + b \to 0 \leq a \operatorname{ch} c + b)$

    Антецеденты обрабатываются проверочными операторами. Уровень срабатывания равен 2.

    $\forall_a(0 \leq \operatorname{th} a + 1)$

    $\forall_a(0 \leq 1 - \operatorname{th} a)$

    Уровень срабатывания равен 3.

59. Целочисленные выражения.

    (a) Натуральный параметр.

        $\forall_{abn}(n -$ натуральное $\& \; 0 < a \;\&\; 0 \leq a + b \to 0 \leq an + b)$

        Выражение a константное, n - переменная. Первый антецедент берется в посылках, второй и третий - обрабатываются проверочными операторами. Уровень срабатывания равен 1.

        $\forall_{Aan}(A(n) \;\&\; \forall_i(A(i) \to a(i) -$ натуральное$) \to 0 \leq a(n))$

        Переменная a - обычная. Второй антецедент берется в посылках, первый - выделен указателем "очевидно", т.е. обрабатывается проверочным оператором, определяемым по виду утверждения $A(n)$. Уровень срабатывания равен 3.

    (b) Наибольший общий делитель.

        $\forall_{ab}(0 \leq$ нод$(a, b))$

        Уровень срабатывания равен 1.

        $\forall_{abc}(\neg(b = 0) \;\&\; a|b \;\&\; a|c \to 0 \leq$ нод$(b, c) - a)$

        Антецеденты обрабатываются проверочными операторами. Уровень срабатывания равен 2.

    (c) Делитель натурального числа.

        $\forall_{mn}(m|n \;\&\; n -$ натуральное $\to 0 \leq n - m)$

        Антецеденты обрабатываются проверочными операторами. Предварительно устанавливается существование посылки с заголовком "натуральное". Уровень срабатывания равен 1.

(d) Целая часть.

$\forall_a(0 \leq a \to 0 \leq [a])$

Антецедент обрабатывается проверочным оператором. Уровень срабатывания равен 1.

$\forall_{mnk}(n - \text{целое}\ \&\ 0 \leq m\ \&\ 0 \leq mn - m + 2k \to 0 \leq m[n/2] + k)$

$\forall_{abc}(0 \leq a\ \&\ ab + c \leq 0 \to a[b] + c \leq 0)$

$\forall_{mnk}(0 \leq k\ \&\ n \leq 0\ \&\ 0 \leq m + nk \to 0 \leq m + n[k])$

$\forall_a(a \leq 0 \to [a] \leq 0)$

Антецеденты обрабатываются проверочными операторами. Уровень срабатывания равен 3.

(e) Положительное целое не меньше единицы.

$\forall_n(n - \text{целое}\ \&\ 0 < n \to 0 \leq n - 1)$

Второй антецедент берется в посылках, первый - обрабатывается проверочным оператором. Уровень срабатывания равен 1.

(f) Конечный отрезок целых чисел.

$\forall_{abmn}(a \in \{m, \ldots, n\}\ \&\ 0 \leq b + m \to 0 \leq a + b)$

$\forall_{abmn}(a \in \{m, \ldots, n\}\ \&\ b + n \leq 0 \to a + b \leq 0)$

Выражение b константное. Первый антецедент берется в посылках, второй - обрабатывается проверочным оператором. Уровень срабатывания приема равен 3.

$\forall_{abmn}(a \in \{m, \ldots, n\}\ \&\ 0 \leq b - n \to 0 \leq b - a)$

Аналогично предыдущему, но выражение b не обязательно константное. Введен сильный ограничитель трудоемкости.

$\forall_a(a \in \{m, \ldots, n\} \to 0 \leq n - a)$

$\forall_a(a \in \{m, \ldots, n\} \to 0 \leq a - m)$

Уровень срабатывания равен 3.

(g) Степень многочлена.

$\forall_a(0 \leq \deg(a))$

Уровень срабатывания равен 1.

$\forall_a(0 \leq \deg(a) - \deg(\text{младшиечлены}(a)))$

Посредством "(младшиечлены(a))" обозначается результат удаления старшего члена многочлена a. Уровень срабатывания равен 2.

(h) Вычет по натуральному модулю.

$\forall_{in}(n - \text{натуральное} \to 0 \leq i (\mathrm{mod}\ n))$

Антецедент обрабатывается проверочным оператором. Уровень срабатывания равен 1.

$\forall_{bin}(n - \text{натуральное}\ \&\ 0 \leq b + 1 \to 0 \leq n - i (\mathrm{mod}\ n) + b)$

Антецеденты обрабатываются проверочными операторами. Уровень срабатывания равен 2.

(i) Сигнум.

$\forall_d(\mathrm{sg}\ d \leq 1)$

Уровень срабатывания равен 1.

$\forall_{abc}(0 \leq a\ \&\ 0 \leq a - |b| \to 0 \leq a + b\ \mathrm{sg}\ c)$

Антецеденты обрабатываются проверочным оператором. Уровень срабатывания равен 2.

(j) Простое число.

$\forall_{pn}(\text{простое}(p)\ \&\ 0 \le m+2 \to 0 \le p+m)$

Первый антецедент берется в посылках, второй - обрабатывается проверочным оператором. Введен сильный ограничитель трудоемкости. Уровень срабатывания равен 4.

(k) Использование неравенства в посылках.

$\forall_{abx}(x - \text{натуральное}\ \&\ a - \text{целое}\ \&\ a < x\ \&\ 0 \le a+b+1 \to 0 \le x+b)$

$\forall_{abx}(x - \text{натуральное}\ \&\ a - \text{целое}\ \&\ x < a\ \&\ 0 \le b-a+1 \to 0 \le b-x)$

Первый и третий антецеденты берутся в посылках, остальные антецеденты обрабатываются проверочными операторами. Вместо заголовка "натуральное" допускается заголовок "целое". Уровень срабатывания равен 3.

$\forall_{abcd}(c \le [a]+d\ \&\ 0 \le b+c-d-1 \to 0 \le a+b)$

Первый антецедент берется в посылках, причем допускается строгое неравенство. Второй антецедент обрабатывается проверочным оператором. Выражение a неконстантное. Уровень срабатывания равен 4.

(l) Дробная часть.

$\forall_{a}(0 \le \text{дробнаячасть}(a))$

Уровень срабатывания равен 1.

60. Физические величины.

(a) Единицы измерения. Теоремы приемов - неравенства, выражающие неотрицательность единиц измерения "см", "дм", "м", "км", "сек", "мин", "час", "дн."(встречающаяся в текстовых задачах единица "день"), "г", "кг", "т", "л", "Н", "Дж", "моль", "руб". Уровень срабатывания равен 1.

(b) Скорость.

$\forall_{ab}(0 \le \text{скорость}(a,b))$

Здесь "скорость(a,b)" - длина векторной скорости точки a в момент b (скорость рассматривается относительно абсолютного пространства). Уровень срабатывания равен 1.

$\forall_{abc}(c = \text{скорость}(a,b) \to 0 \le c)$

Антецедент берется в посылках. Уровень срабатывания равен 3.

(c) Уменьшение высоты при падении тела.

$\forall_{aTpqK}(\text{бросок}(a,T)\ \&\ \text{поверхнземли}(K)\ \&\ p \in T\ \&\ q \in T\ \&\ 0 \le q-p\ \&\ T = [m,n]\ \&\ \text{Скорость}(a,K,m) = \text{вектор0} \to 0 \le \text{крд}(\text{Место}(a,p),K,3) - \text{крд}(\text{Место}(a,q),K,3))$

$\forall_{aTpqK}(\text{бросок}(a,T)\ \&\ \text{поверхнземли}(K)\ \&\ p \in T\ \&\ q \in T\ \&\ 0 \le p-q\ \&\ T = [m,n]\ \&\ \text{Скорость}(a,K,m) = \text{вектор0} \to \text{крд}(\text{Место}(a,p),K,3) - \text{крд}(\text{Место}(a,q),K,3) \le 0)$

Первые два антецедента берутся в посылках. Они означают, что материальная точка a на протяжении периода T движется только под действием силы тяжести, причем движение происходит в окрестности поверхности

Земли. Шестой и седьмой антецеденты тоже берутся в посылках. Первый из них определяет начальный и последний моменты m, n временного промежутка T, второй - указывает, что векторная скорость точки a относительно связанной с поверхностью Земли прямоугольной системы координат K в начальный момент равна нулю. Третий, четвертый и пятый антецеденты обрабатываются проверочными операторами. Выражение "крд(Место(a,t),$K,3)$" обозначает третью $(z-)$ координату в системе K геометрической точки, в которой материальная точка a оказывается в момент t. Уровень срабатывания равен 2.

$\forall_{aTtpqK}(\text{бросок}(a,T)\ \&\ \text{поверхнземли}(K)\ \&\ T=[p,q]\ \&\ t\in T\ \&\ \text{Скорость}(a,K,p)=\text{вектор0}\ \&\ 0\le\text{крд}(\text{Место}(a,q),K,3)\to 0\le\text{крд}(\text{Место}(a,t),K,3))$

Первые три антецедента берутся в посылках. Четвертый и шестой антецеденты обрабатываются проверочными операторами, пятый - выделен указателем "идентификатор". Уровень срабатывания равен 2.

(d) Угловая скорость.

$\forall_{ab}(0\le\text{углскорость}(a,b))$

Здесь "углскорость(a,b)" обозначает модуль угловой скорости материальной точки a в момент b. Уровень срабатывания равен 1.

$\forall_{abc}(c=\text{углскорость}(a,b)\to 0\le c)$

Антецедент берется в посылках. Уровень срабатывания равен 3.

(e) Масса.

$\forall_{a}(0\le\text{масса}(a))$

Уровень срабатывания равен 2.

$\forall_{ab}(b=\text{масса}(a)\to 0\le b)$

Антецедент берется в посылках, уровень срабатывания равен 3.

(f) Длительность процесса.

$\forall_{a}(0\text{длительность}(a))$

Уровень срабатывания равен 1.

$\forall_{ab}(b=\text{длительность}(a)\to 0\le b)$

$\forall_{abc}(\text{длительность}(a)+b=c\to 0\le c-b)$

Антецедент берется в посылках. Уровни срабатывания равны, соответственно, 3 и 4.

(g) Физические константы.

$0\le\text{числоАвогадро}$

$0\le g$

$0\le\text{грав}$

Уровни срабатывания равны 1. Посредством "грав" обозначена гравитационная постоянная.

(h) Исходная длина упругой связи.

$\forall_{a}(0\le\text{исхдлина}(a))$

Уровень срабатывания равен 1.

(i) Атомная масса.

$\forall_{a}(0\le\text{атомнаямасса}(a))$

Уровень срабатывания равен 1.

(j) Количество вещества.

$\forall_{ab}(0 \leq \text{количвещества}(a,b))$

Уровень срабатывания равен 2.

(k) Вертикальная составляющая скорости при падении.

$\forall_{apqtKT}(T = [p,q]$ & $\text{Равноускоренное}(a,K,T)$ & $\text{Скорость}(a,K,p) = \text{вектор0}$ & $0 \leq \text{крд}(\text{Место}(a,p),K,3) - \text{крд}(\text{Место}(a,q),K,3)$ & $t \in T \rightarrow \text{крд}(\text{Скорость}(a,K,t),K,3) \leq 0)$

Материальная точка a движется равноускоренно относительно прямоугольной системы координат K в течение промежутка времени T, причем в начальный момент скорость точки равна 0, а положение ее в последний момент - ниже, чем в начальный. Тогда в произвольный внутренний момент t вертикальная составляющая вектора скорости неположительна. Первые два антецедента берутся в посылках, третий - выделен указателем "идентификатор", четвертый и пятый - обрабатываются проверочными операторами. Уровень срабатывания равен 3.

(l) Направление одномерного движения при торможении.

$\forall_{apqrtKT}(\text{одномерндвиж}(a,K,T)$ & $\text{торможение}(a,T)$ & $T = [p,q]$ & $\text{неподв}(K,T)$ & $\text{вправо}(\text{Скорость}(a,K,p),K)$ & $t \in T$ & $r \in T$ & $0 \leq t - r \rightarrow 0 \leq \text{крд}(\text{Место}(a,t),K,1) - \text{крд}(\text{Место}(a,r),K,1))$

Материальная точка a на промежутке времени T движется по оси абсцисс в неподвижной системе координат K. Суммарная сила, действующая на нее, противоположна скорости движения и равна нулю в случае остановки. В начальный момент рассматриваемого промежутка скорость точки направлена вправо. Тогда в более поздний момент промежутка точка окажется правее, чем в более ранний момент. Первые три антецедента и пятый антецедент берутся в посылках, остальные - обрабатываются проверочными операторами. Уровень срабатывания равен 3.

(m) Вертикальная составляющая ускорения при движении по наклонной плоскости.

$\forall_{abKT}(\text{движениепо}(a,b,T)$ & $\text{Наклплоск}(b,K,c,T)$ & $\text{Силы}(a,\{b,K\},T)$ & $\text{поверхнземли}(K)$ & $\text{неподв}(b,T)$ & $T = [p,q]$ & $\text{Скорость}(a,K,p) = \text{вектор0} \rightarrow \text{крд}(\text{Ускорение}(a,K,T),K,3) \leq 0)$

Материальная точка a движется в течение временного промежутка T по неподвижной наклонной плоскости b под действием только силы реакции со стороны плоскости и силы тяготения. В начальный момент промежутка точка покоилась. Тогда z- составляющая вектора ускорения неположительна. Первые четыре антецедента и шестой антецедент берутся в посылках, остальные - обрабатываются проверочными операторами. Уровень срабатывания равен 3.

(n) Связь вертикальной и горизонтальной составляющей скорости при движении по наклонной плоскости.

$\forall_{abctKT}(\text{движениепо}(a,b,T)$ & $\text{Наклплоск}(b,K,c,T)$ & $\text{неподв}(b,T)$ & $\text{неподв}(K,T)$ & $t \in T$ & $0 \leq c$ & $0 < \text{крд}(\text{Скорость}(a,K,t),K,1) \rightarrow 0 \leq \text{крд}(\text{Скорость}(a,K,t),K,3))$

Материальная точка a движется в течение временного промежутка T по неподвижной наклонной плоскости b, угол наклона c которой неотрицателен. Если в некоторый момент x - координата ее скорости положительна, то в этот же момент z - координата неотрицательна. Первые четыре антецедента и седьмой антецедент берутся в посылках, остальные обрабатываются проверочными операторами. Допускается идентификация седьмого антецедента с нестрогим неравенством. Кроме того, допускается применение приема с перестановкой номеров координат: в антецеденте берется z - координата, а в проверяемом неравенстве - x - координата. Уровень срсбатывания равен 3.

(o) Связь вертикальной и горизонтальной составляющей ускорения при движении по наклонной плоскости.

Аналогично предыдущему.

(p) Направление силы трения.

$\forall_{abcKT}$(движениепо(a, b, T) & $T = [p, q]$ & Наклплоск(b, K, c, T) & неподв(b, T) & поверхнземли(K) & крд(Скорость$(a, K, p), K, 3) = 0$ & $0 < c$ & Силы$(a, \{b, K\}) \to 0 \leq$ крд(силатрения$(a, b, T), K, 1)$)

$\forall_{abcKT}$(движениепо(a, b, T) & $T = [p, q]$ & Наклплоск(b, K, c, T) & неподв(b, T) & поверхнземли(K) & крд(Скорость$(a, K, p), K, 3) = 0$ & $0 < c$ & Силы$(a, \{b, K\}) \to 0 \leq$ крд(силатрения$(a, b, T), K, 3)$)

Материальная точка a движется в течение временного промежутка T по неподвижной наклонной плоскости b, угол наклона которой положителен. На точку действуют только сила реакции со стороны плоскости и сила тяготения. В начальный момент точка покоилась. Тогда x- и z- компоненты силы трения неотрицательны. Первые три антецедента, а также пятый и восьмой антецедент берутся в посылках. Шестой антецедент выделен указателем "идентификатор". Четвертый и седьмой антецеденты обрабатываются проверочными операторами. Уровень срабатывания равен 2.

$\forall_{abtiK}$(движениепо(a, b, t) & неподв(b, t) & $0 <$ крд(Скорость$(a, K, t), K, i) \to$ крд(силатрения$(a, b, t), K, i) \leq 0$)

$\forall_{abtiK}$(движениепо(a, b, t) & неподв(b, t) & крд(Скорость$(a, K, t), K, i) < 0 \to$ $0 \leq$ крд(силатрения$(a, b, t), K, i)$)

Материальная точка a движется в момент t по неподвижной поверхности b. Если в этот момент i-я компонента вектора скорости точки положительна (либо отрицательна), то i-я компонента вектора силы трения неположительна (соответственно, неотрицательна). Первый антецедент берется в посылках, остальные - обрабатываются проверочными операторами. Уровень срабатывания равен 3.

$\forall_{abcKT}$(движениепо(a, b, T) & $T = [p, q]$ & Наклплоск(b, K, c, T) & неподв(b, T) & поверхнземли(K) & $0 <$ крд(Место$(a, p), K, 3) -$ крд(Место$(a, q), K, 3)$ & крд(Скорость$(a, K, q), K, 3) = 0$ & $0 \leq c$ & Силы$(a, \{b, K\}) \to 0 \leq$ крд(силатрения$(a, b, T), K, 1)$)

$\forall_{abcKT}$(движениепо(a, b, T) & $T = [p, q]$ & Наклплоск(b, K, c, T) & неподв(b, T) & поверхнземли(K) & $0 <$ крд(Место$(a, p), K, 3) -$ крд(Место$(a, q), K, 3)$ & крд(Скорость$(a, K, q), K, 3) = 0$ & $0 \leq c$ & Силы$(a, \{b, K\}) \to 0 \leq$ крд(силатрения$(a, b, T), K, 3)$)

Материальная точка a движется в течение временного промежутка T по неподвижной наклонной плоскости b, угол наклона которой неотрицателен. На точку действуют только сила реакции со стороны плоскости и сила тяготения. Высота точки в начальный момент больше ее высоты в последний момент, причем в последний момент точка неподвижна. Тогда x- и z - компоненты силы трения неотрицательны. Первые три антецедента, а также пятый и девятый антецеденты берутся в посылках. Седьмой антецедент выделен указателем "идентификатор". Остальные антецеденты обрабатываются проверочными операторами. Уровень срабатывания равен 2.

$\forall_{abcKT}$(движениепо(a, b, T) & $T = [p, q]$ & Наклплоск(b, K, c, T) & неподв(b, T) & поверхнземли(K) & крд(Место$(a, p), K, 3)$ − крд(Место$(a, q), K, 3) < 0$ & крд(Скорость$(a, K, q), K, 3) = 0$ & $0 \leq c$ & Силы$(a, \{b, K\}) \rightarrow$ крд(силатрения$(a, b, T), K, 1) \leq 0)$

$\forall_{abcKT}$(движениепо(a, b, T) & $T = [p, q]$ & Наклплоск(b, K, c, T) & неподв(b, T) & поверхнземли(K) & крд(Место$(a, p), K, 3)$ − крд(Место$(a, q), K, 3) < 0$ & крд(Скорость$(a, K, q), K, 3) = 0$ & $0 \leq c$ & Силы$(a, \{b, K\}) \rightarrow$ крд(силатрения$(a, b, T), K, 3) \leq 0)$

Аналогично предыдущему, но высота точки в начальный момент меньше ее высоты в последний момент. Тогда x- и z- компоненты силы трения неположительны.

$\forall_{abtiK}$(движениепо(a, b, t) & Наклплоск(b, K, c, t) & Неподв(b, t) & Неподв(K, t) & $0 < c$ & $0 <$ крд(Скорость$(a, K, t), K, 3) \rightarrow$ крд(силатрения$(a, b, t), K, 1) \leq 0)$

$\forall_{abtiK}$(движениепо(a, b, t) & Наклплоск(b, K, c, t) & Неподв(b, t) & Неподв(K, t) & $0 < c$ & крд(Скорость$(a, K, t), K, 3) < 0 \rightarrow 0 \leq$ крд(силатрения$(a, b, t), K, 1))$

$\forall_{abtiKT}$(движениепо(a, b, T) & Наклплоск(b, K, c, T) & неподв(b, T) & Неподв(K, t) & $0 < c$ & $0 <$ крд(Скорость$(a, K, t), K, 3)$ & $t \in T \rightarrow$ крд(силатрения$(a, b, t), K, 1) \leq 0)$

Материальная точка a движется по неподвижной наклонной плоскости b, имеющей положительный угол наклона. Если z - компонента вектора ее скорости положительна (отрицательна), то x - компонента силы трения неположительна (неотрицательна). Первые два антецедента берутся в посылках, остальные - обрабатываются проверочными операторами. Уровень срабатывания равен 3.

$\forall_{abpKT}$(движениепо(a, b, T) & Наклплоск(b, K, p, T) & $0 < p$ & поднимается(a, K, T) & неподв$(b, T) \rightarrow$ крд(силатрения$(a, b, T), K, 1) \leq 0)$

$\forall_{abpKT}$(движениепо(a, b, T) & Наклплоск(b, K, p, T) & $0 < p$ & поднимается(a, K, T) & неподв$(b, T) \rightarrow$ крд(силатрения$(a, b, T), K, 3) \leq 0)$

Материальная точка a поднимается в течение промежутка времени T по неподвижной наклонной плоскости b, имеющей положительный угол наклона. Тогда x- и z - координаты средней силы трения неположительны. Первый, второй и четвертый антецеденты берутся в посылках, остальные - обрабатываются проверочными операторами. Уровень срабатывания равен 3.

$\forall_{abtiK}$(движениепо(a, b, T) & Наклплоск(b, K, c, T) & неподв(b, T) & неподв(K, T) & $0 < c$ & $T = [p, q]$ & Скорость$(a, K, p) =$ вектор0 & Равноускоренное(a, K, T) & крд(Место$(a, q), K, 3)$ − крд(Место$(a, p), K, 3) < 0 \rightarrow 0 \leq$ крд(силатрения$(a, b, T), K, 1))$

$\forall_{abtiK}$(движениепо(a, b, T) & Наклплоск(b, K, c, T) & неподв(b, T) & неподв(

$K,T)$ & $0 < c$ & $T = [p,q]$ & Скорость(a,K,p) = вектор0 & Равноускоренное(a,K,T) & крд(Место$(a,q),K,3)$ − крд(Место$(a,p),K,3) < 0 \to 0 \leq$ крд(силатрения$(a,b,T),K,3)$

Материальная точка a движется в течение периода T по неподвижной наклонной плоскости b, имеющей положительный угол наклона. Движение равноускоренное, причем в начальный момент точка покоится. Высота точки в последний момент меньше ее высоты в начальный момент. Тогда x- и z- компоненты силы трения неотрицательны. Первый, второй, шестой, седьмой и восьмой антецеденты берутся в посылках, остальные - обрабатываются проверочными операторами. Уровень срабатывания равен 3.

(q) Направление силы взаимодействия между телами, соединенными гибкой связью.

$\forall_{abctK}$(гибкаясвязь(a,b,c,t) & одномерндвиж(a,K,t) & одномерндвиж(b,K,t) & $0 \leq$ крд(Место$(a,t),K,1)$−крд(Место$(b,t),K,1) \to$ крд(сила$(a,b,t),K,1) \leq 0)$

$\forall_{abctK}$(гибкаясвязь(a,b,c,t) & одномерндвиж(a,K,t) & одномерндвиж(b,K,t) & $0 \leq$ крд(Место$(a,t),K,1)$ − крд(Место$(b,t),K,1) \to 0 \leq$ крд(сила$(a,b,t),K,1))$

Материальные точки a,b в момент t соединены тонкой нерастяжимой нитью c и движутся вдоль оси абсцисс. Точка a находится правее точки b. Тогда x- координата вектора силового воздействия точки b на точку a неположительна, а точки a на точку b - неотрицательна. Первые три антецедента берутся в посылках, четвертый - обрабатывается проверочным оператором. Уровень срабатывания равен 3.

(r) Коэффициент трения.

$\forall_{ab}(0 \leq$ коэффтрения$(a,b))$

Уровень обращения равен 2.

(s) Удлинение упругой связи.

$\forall_{at}$(шнур$(a) \to 0 \leq$ удлинсвязи$(a,t))$

Удлинение гибкой упругой нити неотрицательно. Антецедент берется в посыках, уровень срабатывания равен 1.

$\forall_{abptABKT}$(упругсвязь(a,b,p,T) & одномерный(сила$(a,b,t),K)$ & $0 \leq$ крд(сила$(a,b,t),K,1)$ & Место$(a,t) = A$ & Место$(b,t) = B$ & $0 \leq$ крд$(B,K,1)$ − крд$(A,K,1)$ & $t \in T$ & прямкоорд$(K) \to 0 \leq$ удлинсвязи$(p,t))$

Материальные точки a,b на временном промежутке T соединены упругой (не обязательно гибкой) связью p, причем вектор силового воздействия b на a направлен вправо вдоль оси абсцисс. Если в момент t промежутка T точка a находится левее точки b, то удлинение связи в этот момент неотрицательно. Первый и последний антецеденты берутся в посылках, четвертый и пятый - выделены указателем "идентификатор". Остальные антецеденты обрабатываются проверочными операторами. Уровень срабатывания равен 3.

(t) Направление силы, поднимающей тело вверх по наклонной плоскости.

$\forall_{abcKT}$(движениепо(a,b,T) & Наклплоск(b,K,d,T) & $0 < d$ & неподв(b,T) & поверхнземли(K) & тянет(c,a,T) & поднимается$(a,K,T) \to 0 \leq$ крд(сила$(a,c,T),K,3))$

$\forall_{abcKT}$(движениепо(a,b,T) & Наклплоск(b,K,d,T) & $0<d$ & неподв(b,T) & поверхнземли(K) & тянет(c,a,T) & поднимается$(a,K,T)\rightarrow 0\leq$ крд(сила$(a,c,T),K,1)$)

Материальную точку a тянет вверх по неподвижной наклонной плоскости b, имеющей положительный угол наклона, некоторая внешняя сила. Тогда x- и z- компоненты вектора этой силы неотрицательны. Третий и четвертый антецеденты обрабатываются проверочными операторами, остальные - берутся в посылках. Уровень срабатывания равен 3.

(u) Направление ускорения.

$\forall_{apqKT}$(движвправо(a,K,T) & Равноускоренное(a,K,T) & $T=[p,q]$ & Скорость$(a,K,q)=$ вектор0 $\rightarrow$ крд(Ускорение$(a,K,T),K,1)\leq 0$)

Материальная точка a движется на промежутке времени T равноускоренно вправо. В последний момент промежутка она останавливается. Тогда x - компонента вектора ускорения неположительна. Все антецеденты берутся в посылках. Уровень срабатывания равен 3.

61. Символы бесконечности.

$\forall_a(a-$ число $\rightarrow -\infty\leq a)$

$\forall_a(a-$ число $\rightarrow a\leq\infty)$

$\forall_a(a-$ число $\rightarrow a\leq$ счетное)

$\forall_a(a-$ число $\rightarrow a\leq$ континуум)

счетное $\leq$ континуум

Антецедент обрабатывается проверочным оператором. Уровень срабатывания равен 1.

62. Скалярное произведение.

$\forall_a(0\leq$ скалумнож$(a,a))$

Уровень срабатывания равен 1.

$\forall_{ABCDn}(0\leq(l(AB))^n(l(CD))^n-($скалумнож(вектор$(AB)$, вектор$(CD)$)$)^n)$

Переменная n идентифицируется с натуральной константой. Уровень срабатывания равен 3.

63. Угол между вектором и осью абсцисс.

$\forall_{aABCDK}(0\leq$ крд$(v,K,1)$ & $K=(A,B,C,D)$ & прямкоорд$(K)\rightarrow 0\leq\pi/2-$ уголмежду$(v,$ вектор$(AB)))$

Если x- координата вектора в прямоугольной системе координат K неотрицательна, то угол между этим вектором и направляющим вектором оси абсцисс не превосходит половины пи. Первый антецедент обрабатывается проверочным оператором, остальные - берутся в посылках. Уровень срабатывания равен 3.

64. Координата вектора.

(a) Вектор направлен вверх.

$\forall_{vK}$(верхнапр$(v,K)\rightarrow 0\leq$ крд$(v,K,3))$

Антецедент представляет собой переформулировку условия неотрицательности z - компоненты вектора. Он берется в посылках. Уровень срабатывания равен 3.

$\forall_{aK}(\text{вверх}(a,K) \to 0 \leq \text{крд}(a,K,3))$

Вектор направлен вертикально вверх. Антецедент обрабатывается проверочным оператором. Уровень срабатывания равен 1.

$\forall_{axK}(\text{вверх}(a,K)\ \&\ \text{крд}(a,K,3) = x \to 0 \leq x)$

Антецеденты берутся в посылках. Введен ускоряющий фильтр, проверяющий наличие посылки с заголовком "вверх". Уровень срабатывания равен 4.

(b) Вектор направлен вниз.

$\forall_{aKx}(\text{вниз}(a,K) \to \text{крд}(a,K,3) \leq 0)$

Антецедент обрабатывается проверочным оператором. Уровень срабатывания равен 2.

$\forall_{abxK}(\text{вниз}(a,K)\ \&\ \text{крд}(a,K,3) = bx\ \&\ b < 0 \to 0 \leq x)$

Первые два антецедента берутся в посылках, третий - обрабатывается проверочным оператором. x - переменная. Используется ускоряющий фильтр - наличие посылки с заголовком "вниз". Уровень срабатывания равен 1.

(c) Вектор направлен вправо.

$\forall_{vK}(\text{вправо}(v,K) \to 0 \leq \text{крд}(v,K,1))$

Антецедент обрабатывается проверочным оператором. Уровень срабатывания равен 3.

(d) Вектор направлен влево.

$\forall_{vK}(\text{влево}(v,K) \to \text{крд}(v,K,1) \leq 0)$

Аналогично предыдущему.

65. Мощность множества.

$\forall_{Ab}(\text{card}A = b \to 0 \leq b)$

Антецедент берется в посылках. Введен ускоряющий фильтр, проверяющий наличие посылки с символом "мощность". Уровень срабатывания равен 3.

$\forall_{A}(0 \leq \text{card}A)$

Уровень срабатывания равен 2.

$\forall_{fnyAB}(\text{Отображение}(f,A,B)\ \&\ y \in B\ \&\ \text{card}A = n \to 0 \leq n - \text{card}(\text{слой}(f,y)))$

Мощность прообраза элемента не превосходит мощности области определения отображения. Первый антецедент берется в посылках, второй - обрабатывается проверочным оператором, третий - выделен указателем "идентификатор". Он использует для определения мощности множества A нормализатор "нормможность". Уровень срабатывания равен 2.

$\forall_{ab}(a \subseteq b \to 0 \leq \text{card}\ b - \text{card}\ a)$

Антецедент берется в посылках. Указатели "смравно" разрешают косвенную идентификацию подтермов $\text{card}\ a, \text{card}\ b$ с использованием равенств для них, имеющихся в посылках. Ускоряющий фильтр проверяет наличие посылки с заголовком "содержится". Уровень срабатывания равен 3.

66. Длина набора.

    $\forall_f(0 \le l(f))$

    Здесь $l(f)$ - прорисовываемое формульным редактором выражение "длинанабора(f)". Уровень срабатывания равен 1.

    $\forall_{An}(n = l(A) \to 0 \le n)$

    Антецедент берется в посылках. Уровень срабатывания равен 3.

67. Вероятность.

    $\forall_{AB}(0 \le \text{вероятность}(A, B))$

    $\forall_{ABC}(0 \le \text{услвероятн}(A, B, C))$

    Посредством "вероятность(A, B)" обозначена вероятность события A в вероятностном пространстве B, "услвероятн(A, B, C)" - условная вероятность события A при наличии события B в пространстве C. Уровень срабатывания равен 1.

68. Случайные величины.

    $\forall_{abfgxP}(0 \le b \;\&\; 0 \le a - b \;\&\; f = \text{функраспред}(g, P) \to 0 \le a - bf(x))$

    Значение функции распределения случайной величины не превосходит 1. Первые два антецедента обрабатываются проверочными операторами, третий - берется в посылках. Переменная f - обычная, т.е. $f(x)$ идентифицируется с термом вида "значение(ft)". Уровень срабатывания равен 4.

    $\forall_{fgxP}(f = \text{функраспред}(g, P) \to 0 \le f(x))$

    Антецедент берется в посылках. Уровень срабатывания равен 4.

    $\forall_{fgxP}(f = \text{плотнраспред}(g, P) \to 0 \le f(x))$

    Аналогично предыдущему.

69. Двоичные наборы.

    $\forall_{anx}(\text{двнабор}(x, n) \to 0 \le n - \text{колич}(x, a))$

    Количество разрядов двоичного набора x, равных заданному значению a, не превосходит длины набора. Антецедент берется в посылках. Уровень срабатывания равен 3.

    $\forall_{ab}(0 \le \text{дврасст}(a, b))$

    Посредством дврасст(a, b) обозначено расстояние по Хэммингу между двоичными наборами a, b. Уровень срабатывания равен 1.

    $\forall_{abcd}(0 \le d \;\&\; 0 \le c + d \to 0 \le c\text{двнорма}(a \;\&\; b) + d\text{двнорма}(a))$

    $\forall_{abcd}(0 \le \;\&\; c + d \le 0 \to c\text{двнорма}(a \;\&\; b) + d\text{двнорма}(a) \le 0)$

    Число единиц в конъюнкции двоичных наборов a, b не превосходит числа единиц в наборе a. Антецеденты обрабатываются проверочными операторами. Уровень срабатывания равен 3.

70. Точки минимума либо максимума.

    $\forall_{abcmnk}(\text{Pmax}(\lambda_i(a(i), i \in \{m, \dots, n\}), b, c) \;\&\; 0 \le m + k \to 0 \le b + k)$

    Утверждение Pmax(f, b, c) означает, что b есть точка, в которой функция f принимает наибольшее значение c на своей области определения. В данном случае

f определена на конечном отрезке целых чисел $\{m, \ldots, n\}$, так что b не менее m. Первый антецедент берется в посылках, второй - обрабатывается проверочным оператором. Введен указатель "вариант(...)", разрешающий идентификацию первого антецедента с утверждением вида Pmin(...). Уровень срабатывания равен 3.

$\forall_{abcmnk}(\mathrm{Pmax}(\lambda_i(a(i), i \in \{m, \ldots, n\}), b, c) \,\&\, 0 \leq k - n \rightarrow 0 \leq k - b)$

Аналогично предыдущему.

71. Разное.

    (a) Монотонность логарифма.

    $\forall_{axy}(0 < a - 1 \,\&\, x \leq y \rightarrow \log_a x \leq \log_a y)$

    $\forall_{axy}(0 < -a + 1 \,\&\, y \leq x \rightarrow \log_a x \leq \log_a y)$

    Антецеденты обрабатываются проверочными операторами. Уровень срабатывания равен 2.

    (b) Число вершин порожденного подграфа не превосходит числа вершин исходного графа.

    $\forall_{ABc}$(порождподграф(A, B) & $0 \leq c-\mathrm{card}$(вершины$(A)) \rightarrow 0 \leq c-\mathrm{card}$(вершины$(B)))$

    Первый антецедент берется в посылках, второй - обрабатывается проверочным оператором. Уровень срабатывания равен 2.

    (c) Значение перестановки конечного отрезка целых чисел заключено между границами этого отрезка.

    $\forall_{fmni}$(перестановка$(f, \{m, \ldots, n\}) \rightarrow 0 \leq n - f(i))$

    $\forall_{fmni}$(перестановка$(f, \{m, \ldots, n\}) \rightarrow 0 \leq f(i) - m)$

    Антецедент берется в посылках. Уровень срабатывания равен 3.

Проверочный оператор "провменьшеилиравно"

Оператор аналогичен оператору "провменьше". Поэтому ограничиваемся перечислением названием групп его приемов. Если некоторые приемы группы сколь-нибудь сильно отличаются от соответствующих приемов оператора "провменьше", приводим их теоремы.

1. Обращение к оператору "усмменьшеилиравно".

2. Поглощение одного слагаемого двумя с применением квадратичной оценки.

   $\forall_{abcdefghij}(0 \leq a \,\&\, 0 \leq b \,\&\, c^2 = ab \,\&\, d = 2e - |f| \,\&\, g = 2h - |f| \,\&\, 0 \leq d \,\&\, 0 \leq g \,\&\, 2j \leq ad + bg + 2i \rightarrow j \leq ea + hb + fc + i)$

   Переменные e, f, h идентифицируются с десятичными константами. Первые два антецедента обрабатываются проверочными операторами, третий - выделен указателем "идентификатор". Антецеденты с 4-го по 7 выделены указателем "программа". Восьмой антецедент, выделенный указателем "проверка", выполняет рекурсивное обращение к тому же оператору. Уровень срабатывания приема равен 1.

   $\forall_{abcdefghi}(0 \leq a \,\&\, 0 \leq b \,\&\, c^2 = ab \,\&\, 0 < d \,\&\, 0 < e \,\&\, f = 4de - g^2 \,\&\, 0 \leq f \,\&\, 4di \leq bf + 4dh \rightarrow i \leq da + eb + gc + h)$

   Аналогично случаю строгих неравенств.

3. Поглощение части одного слагаемого двумя слагаемыми с применением квадратичной оценки.

   $\forall_{abcdefghi}(0 \leq a \,\&\, 0 \leq b \,\&\, c^2 = ab \,\&\, d^2 = ef \,\&\, 0 < e \,\&\, 0 < f \,\&\, i \leq (g+2d)c+h \rightarrow i \leq ea + fb + gc + h)$

4. Поглощение одного слагаемого двумя с двукратным использованием одного из них.

5. Поглощение одного слагаемого тремя с применением квадратичной оценки. Аналогично случаю строгих неравенств, но уровень срабатывания равен 4.

6. Поглощение одного слагаемого тремя с применением кубической оценки.

7. Поглощение части одного слагаемого тремя слагаемыми.

8. Поглощение одного слагаемого двумя с применением кубической оценки.

9. Извлечение нижней оценки произведения из неравенств для сомножителей.

10. Извлечение нижней оценки суммы квадратов из неравенства для суммы.

    $\forall_{abcdefg}(0 \leq f \,\&\, 0 \leq ad + bd + cd + e \,\&\, 0 < d \,\&\, 0 \leq fe^2 + 3d^2g \,\&\, e \leq 0 \rightarrow 0 \leq a^2f + b^2f + c^2f + g)$

    Второй антецедент берется в посылках, причем допускается случай строгого неравенства. Четвертый антецедент реализует рекурсивное обращение к процедуре "провменьшеилиравно". Остальные антецеденты обрабатываются проверочными операторами. Уровень срабатывания равен 2.

    $\forall_{abcdef}(0 \leq e \,\&\, 0 \leq ad+bd+c \,\&\, 0 < d \,\&\, 0 \leq ec^2+2d^2f \,\&\, c \leq 0 \rightarrow 0 \leq a^2e+b^2e+f)$

    Аналогично случаю строгих неравенств.

11. Оценка для суммы квадратов через сумму, позволяющая свести правую часть неравенства к виду разности квадратов.

12. Попытка замены суммы квадратов на удвоенное произведение.

13. Попытка замены суммы квадрата и удвоенного произведения на другой квадрат, взятый с обратным знаком.

14. Оценка сверху для радикала второй степени.

15. Оценка, использующая выпуклость степенной функции.

16. Попытка группировки двух слагаемых.

17. Использование неравенства из посылок, дающего константную оценку для переменной.

18. Возведение в квадрат неравенства с одним радикалом.

Нормализатор решения нестрогих неравенств "уравнменьшеилиравно"

Оператор аналогичен нормализатору решения строгих неравенств "уравнменьше", и далее ограничиваемся перечислением названий групп приемов:

1. Устранение минуса в неравенстве с нулевой частью.
2. Устранение дроби в неравенстве с нулевой частью.
3. Устранение степени в неравенстве с нулевой частью.
4. Неравенство с произведением в одной части и нулем в другой.
5. Неравенство с неизвестной дробью в одной части.
6. Решение неравенств $-x \leq a, a \leq x$.
7. Решение неравенств $x + a \leq b, b \leq x + a$.
8. Решение неравенств $xa \leq b, b \leq xa$.
9. Решение неравенств с модулем.
10. Группировка неизвестных слагаемых в одной части.
11. Решение простейшего степенного неравенства с положительным показателем степени.
12. Решение простейшего степенного неравенства с отрицательным показателем степени.
13. Попытка разложения на множители разности частей неравенства.
14. Усмотрение противоречия либо равенства из двух противоположных неравенств для неизвестного выражения.
15. Попытка раскрывания скобок с неизвестными слагаемыми.
16. Квадратные неравенства.
17. Группировка слагаемых с одинаковым неизвестным радикалом.
18. Возведение в квадрат неравенства с одним либо двумя радикалами.
19. Потенцирование логарифмических неравенств.
20. Логарифмирование показательных неравенств.
21. Переход к тангенсу половинного угла в тригонометрических неравенствах.
22. Линейная комбинация синуса и косинуса в неизвестной части неравенства.
23. Переход от неравенства для синуса к уравнению.

    $\forall_x (1 \leq \sin x \leftrightarrow \sin x = 1)$

    $\forall_x (\sin x \leq -1 \leftrightarrow \sin x = -1)$

    Уровень срабатывания равен 1.

24. Разное.
    (a) Усмотрение истинности неравенства для суммы квадратов и удвоенного произведения, домноженного на неизвестный коэффициент.

        $\forall_{abx}(x \leq 1 \ \& \ -1 \leq x \rightarrow 2abx \leq a^2 + b^2)$

        Антецеденты берутся в посылках. Переменная x идентифицируется с неизвестной. Уровень срабатывания равен 1.
    (b) Усмотрение квадратного неравенства в показательном неравенстве с известными основаниями неизвестных степеней.

        $\forall_{abcdefgh}(2e - c = 0 \rightarrow ab^{c+d} + gb^{e+f} \leq h \leftrightarrow ab^c b^d + gb^e b^f \leq h)$

        Выражения a, b, d, f, g, h, не содержат неизвестных, выражения c, e - содержат. Допускается перестановка левой и правой частей. В результате предпринимаемой приемом перегруппировки возникает квадратное неравенство относительно b^e. Уровень срабатывания равен 1.

Нормализатор нестрогих неравенств "нормменьшеилиравно"

Оператор аналогичен нормализатору строгих неравенств. Он имеет следующие группы приемов:

1. Устранение минуса в неравенстве с нулевой противоположной частью.
2. Устранение множителя определенного знака в неравенстве с нулевой противоположной частью.
3. Устранение дроби в неравенстве с нулевой частью.
4. Устранение степени в неравенстве с нулевой частью.
5. Усмотрение истинности неравенства при помощи оператора "усмменьшеилиравно".
6. Усмотрение ложности неравенства при помощи оператора "усмменьше".
7. Возведение в квадрат частей неравенства с неизвестными радикалами.
8. Усмотрение равенства.

   $\forall_a(0 \leq a \rightarrow a \leq 0 \leftrightarrow a = 0)$

   Уровень срабатывания равен 1.

Нормализатор нестрогих неравенств для параметров "стандменьшеилиравно"

Оператор аналогичен нормализатору строгих неравенств для параметров "стандменьше".

Синтезатор "верхняяоценка"

Если необходимо получить нестрогую верхнюю либо нижнюю оценку некоторого выражения относительно заданного списка посылок, то используются синтезаторы "верхняяоценка" и "нижняяоценка". Получение строгих оценок пока не предусмотрено. Антецедент теоремы приема, обращающийся к синтезатору "верхняяоценка", имеет вид $a \leq x$. Здесь a - некоторое выражение, все переменные которого к моменту обращения уже идентифицированы; x - еще не идентифицированная переменная, которой присваивается после работы синтезатора найденная им верхняя оценка для a. Теорема приема синтезатора имеет своим консеквентом утверждение вида $A \leq B$, где A - оцениваемое выражение, идентифицируемое с поступающим на вход синтезатора выражением, B - верхняя оценка для A, присваиваемая выходной переменной. Если при обращении синтезатору передается комментарий (независит $x_1 \dots x_n$), то верхняя оценка представляет собой некоторое выражение, не содержащее переменных $x_1, \dots x_n$. В противном случае оценка является константным выражением. Результаты обращения к синтезатору сохраняются в буфере, откуда и извлекаются при повторном идентичном обращении. Синтезаторы "верхняяоценка", "нижняяоценка" - перечисляющие операторы. Они выдают не какую-то одну найденную их приемами оценку, а всевозможные такие оценки, избегая при перечислении повторений. Синтезатор "верхняяоценка" имеет следующие приемы:

1. Синус и косинус.

   $\forall_a(|\cos a| \leq 1)$

   $\forall_a(|\sin a| \leq 1)$

   $\forall_a(\cos a \leq 1)$

   $\forall_a(\sin a \leq 1)$

   Уровень срабатывания равен 2.

2. Линейная комбинация синуса и косинуса.

   $\forall_{abc}(a \sin b + c \cos b \leq \sqrt{a^2 + c^2})$

   $\forall_{abc}(|a \sin b + c \cos b| \leq \sqrt{a^2 + c^2})$

   Переменные a, c идентифицируются с константными выражениями. Уровень срабатывания равен 1.

3. Трехчлен с синусом и косинусом.

   $\forall_{abc}(a(\sin x)^2 + b(\cos x)^2 + c \sin x \cos x \leq (a + b + \sqrt{(a-b)^2 + c^2})/2)$

   Выражения a, b, c константные. Уровень срабатывания равен 1.

4. Плюс.

   $\forall_{abcd}(a \leq c \;\&\; b \leq d \rightarrow a + b \leq c + d)$

   Антецеденты выполняют рекурсивные обращения к синтезатору "верхняяоценка". Указатель "первыйтерм(фикс(0 1 1))" обеспечиает одновременную обработку всех слагаемых для суммы произвольной длины. Уровень срабатывания равен 2.

   $\forall_{abc}(a \leq b \rightarrow a + c \leq b + c)$

Выражение c константное. Антецедент выполняет рекурсивное обращение к оператору "верхняяоценка". Указатель "спуск" блокирует альтернативные попытки получения верхней оценки. Уровень срабатывания равен 1.

5. Минус.

   $\forall_{abc}(c \leq a + b \to -a - b \leq -c)$

   Антецедент обращается к синтезатору "нижняяоценка". Уровень срабатывания равен 1.

   $\forall_{ab}(b \leq a \to -a \leq -b)$

   Антецедент обращается к синтезатору "нижняяоценка". Уровень срабатывания равен 2.

   $\forall_{ab}(|a| \leq b \to -a \leq b)$

   Выражение a имеет вхождение тригонометрической операции. Антецедент выполняет рекурсивное обращение к оператору "верхняяоценка". Уровень срабатывания равен 3.

6. Умножение.

   $\forall_{abcd}(|a| \leq b \;\&\; |c| \leq d \to |ac| \leq bd)$

   Антецеденты обращаются к оператору "верхняяоценка". Уровень срабатывания равен 1.

   $\forall_{abc}(0 < a \;\&\; b \leq c \to ab \leq ac)$

   Переменная a идентифицируется с произведением всех константных сомножителей. Это произведение отлично от единицы. Первый антецедент обрабатывается проверочным оператором, второй - оператором "верхняяоценка". Указатель "спуск" блокирует альтернативные попытки получения верхней оценки. Уровень срабатывания равен 2.

   $\forall_{abcd}(0 \leq a \;\&\; a \leq c \;\&\; b \leq d \;\&\; 0 \leq d \to ab \leq cd)$

   Выражения a, b неконстантные и не являются произведениями. Второй и третий антецеденты обращаются к оператору "верхняяоценка". Первый и последний антецеденты обрабатываются проверочными операторами. Уровень срабатывания равен 3.

7. Дробь.

   $\forall_{abc}(0 < c \;\&\; a \leq b \to a/c \leq b/c)$

   Выражение c константное. Первый антецедент обрабатывается проверочным оператором, второй - оператором "верхняяоценка". Указатель "спуск" блокирует альтернативные попытки получения верхней оценки. Уровень срабатывания равен 2.

   $\forall_{abcd}(0 < a \;\&\; b/c \leq d \to ab/c \leq ad)$

   Выражение a константное. Первый антецедент обрабатывается проверочным оператором, второй - обращается к оператору "верхняяоценка". Уровень срабатывания равен 2.

   $\forall_{ab}(0 < a \;\&\; 0 \leq b \to ab/(a^2 + b^2) \leq 1/2)$

Антецеденты обрабатываются проверочными операторами. Уровень срабатывания равен 1.

$\forall_{abc}(0 < c \ \& \ c \leq b \ \& \ 0 < a \to a/b \leq a/c)$

Выражение a константное. Первый и третий антецеденты обрабатываются проверочными операторами, второй - оператором "нижняяоценка". Уровень срабатывания равен 4.

$\forall_{abcdex}(0 < ad - bc \ \& \ 0 < cx + d \ \& \ a/c \leq e \to (ax + b)/(cx + d) \leq e)$

x - переменная, не входящая в выражения a, b, c, d. Первые два антецедента обрабатываются проверочными операторами, третий - оператором "верхняяоценка". Уровень срабатывания равен 1.

$\forall_{abc}(0 < c \ \& \ c < b \ \& \ 0 < a \to a/b \leq a/c)$

Выражения a, c константные. Второй антецедент берется в посылках, первый и третий - обрабатываются проверочными операторами. Уровень срабатывания равен 4.

$\forall_{abcdft}(0 \leq c - f(t) \ \& \ ac/b \leq d \ \& \ 0 < a \ \& \ 0 < b \to af(t)/b \leq d)$

Переменная f - обычная. Имеется комментарий (независит ...), перечисляющий переменные, которые не долюны входить в верхнюю оценку. Эти переменные встречаются как в t, так и в b. Первый антецедент берется в посылках, второй - обращается к оператору "верхняяоценка" (комментарий "независит" передается при обращении автоматически). Последние два антецедента обрабатываются проверочными операторами. Уровень срабатывания равен 3.

$\forall_{mnp}(0 < k \ \& \ 0 \leq n - k \ \& \ m \leq p \to m/n \leq |p|/k)$

Имеется комментарий (независит ...), переменные которого встречаются как в m, так и в n. Второй антецедент берется в посылках, причем допускается случай строгого неравенства. Переменные комментария "независит" не входят в k. Первый антецедент обрабатывается проверочным оператором, последний - оператором "верхняяоценка". Уровень срабатывания равен 4.

$\forall_{abc}(0 < c \ \& \ c \leq b \to a/b \leq |a|/c)$

Имеется комментарий (независит ...), переменные которого не входят в a. Первый антецедент обрабатывается проверочным оператором, второй - оператором "нижняяоценка". Уровень срабатывания равен 4.

8. Степень.

   $\forall_{abc}(0 < c \ \& \ |a| \leq b \to a^c \leq b^c)$

   $\forall_{abc}(0 < c \ \& \ |a| \leq b \to |a^c| \leq b^c)$

   Выражение c константное. Первый антецедент обрабатывается проверочным оператором, второй - оператором "верхняяоценка". Уровень срабатывания равен 2.

   $\forall_{a}((-1)^a \leq 1)$

   Уровень срабатывания равен 3.

   $\forall_{abc}(c - \mathrm{rational} \ \& \ \neg(\text{числитель}(c) - \mathrm{even}) \ \& \ \neg(\text{знаменатель}(c) - \mathrm{even}) \ \& \ a \leq b \to a^c \leq b^c)$

Первые три антецедента обрабатываются проверочными операторами, последний - оператором "верхняяоценка". Уровень срабатывания равен 1.

$\forall_{abc}(0 < c \ \& \ a \leq b \ \& \ 0 \leq a \to a^c \leq b^c)$

$\forall_{abcd}(0 < c \ \& \ a/d \leq b \ \& \ 0 \leq a \ \& \ 0 \leq d \to a^c/d^c \leq b^c)$

Выражение c константное. Второй антецедент обрабатывается оператором "верхняяоценка", остальные антецеденты обрабатываются проверочными операторами. Уровень срабатывания равен 1.

9. Константа.

   $\forall_a(a \leq a)$

   Выражение a константное. Оно берется в качестве собственной верхней оценки. Уровень срабатывания равен 1. Создана еще одна версия приема, в которой требуется наличие комментария (параметры ...), не содержащего переменных выражения a. Здесь выражение a берется в качестве собственной верхней оценки на том основании, что не содержит варьируемых переменных.

10. Использование посылок.

    (a) Неравенство с нулевой частью.

    $\forall_{ab}(0 < b - a \to a \leq b)$

    $\forall_{ab}(a + b < 0 \to a \leq -b)$

    Выражение b константное. Антецедент берется в посылках, причем допускается нестрогое неравенство. Уровень срабатывания равен 1.

    (b) Неравенство для оцениваемого выражения.

    $\forall_{ab}(a < b \to a \leq b)$

    Выражение b константное. Антецедент берется в посылках, причем допускается случай нестрогого неравенства. Уровень срабатывания равен 1.

    (c) Транзитивность.

    $\forall_{abcd}(0 \leq -a + c \ \& \ c \leq d \to a \leq d)$

    Первый антецедент берется в посылках, второй - обрабатывается оператором "верхняяоценка". Выражения a, c не имеют общих параметров, при этом первое из них не является суммой. Уровень срабатывания равен 3.

    $\forall_{abcx}(a + c \leq x \ \& \ 0 \leq b + c \to a - b \leq x)$

    Выражение b неконстантное. Имеется комментарий (независит ...). Второй антецедент берется в посылках, первый - обрабатывается оператором "верхняяоценка". Уровень срабатывания равен 3.

    (d) Оценка модуля с помощью двух противоположных неравенств.

    $\forall_{abc}(0 \leq a + c \ \& \ 0 \leq b - c \to |c| \leq \max(|a|, |b|))$

    Выражения a, b константные. Оба антецедента берутся в посылках, причем допускаются строгие неравенства. Уровень срабатывания равен 2.

    (e) Оценка модуля с помощью неравенства для модуля.

    $\forall_{abcemnpq}(\neg(m = 0) \ \& \ 0 < e - |am + n| \ \& \ |b| \leq q \ \& \ |cm - bn| \leq p \to |ab + c| \leq (p + qe)/|m|)$

    Второй антецедент берется в посылках, первый - обрабатывается проверочным оператором, третий и четвертый - обрабатываются оператором "верхняяоценка". Имеется комментарий (независит ...), переменные которого входят в a и не входят в e, m. Уровень срабатывания равен 3.

(f) Оценка с помощью неравенства для модуля.

$\forall_{ab}(0 < a - |b| \rightarrow b \leq |a|)$

Антецедент берется в посылках, причем допускается нестрогое неравенство. Имеется комментарий (независит ...), переменные которого не входят в a. Уровень срабатывания равен 1.

(g) Оценка величины с помощью неравенства для отклонения этой величины.

$\forall_{abcd}(\neg(d = 0) \ \& \ 0 < a - |b + cd| \rightarrow c \leq -b/d + a/|d|)$

Второй антецедент берется в посылках, причем допускается случай нестрогого неравенства. Первый антецедент обрабатывается проверочным оператором. Имеется комментарий (независит ...), переменные которого входят в c и не входят в a, b, d. Уровень срабатывания равен 1.

(h) Оценка значений функции натурального аргумента на конечном промежутке.

$\forall_{mn}(n$ – натуральное & m – целое & $0 \leq m - n \rightarrow f(n) \leq \sup(\text{set}_x(\exists_i(i \in \{1, \ldots, m,\} \ \& \ x = f(i)))))$

Первые два антецедента обрабатываются проверочными операторами, третий - берется в посылках. Имеется комментарий (независит ...), переменные которого входят в n и не входят в m. Уровень срабатывания равен 3.

11. Квадратный трехчлен.

$\forall_{abcx}(a < 0 \rightarrow ax^2 + bx + c \leq c - b^2/4a)$

Выражения a, b, c константные. Антецедент обрабатывается проверочным оператором. Уровень срабатывания равен 1.

12. Кубический четырехчлен на отрезке.

$\forall_{abcdefpqr}(p = b^2 - 3ac \ \& \ 0 \leq p \ \& \ \neg(a = 0) \ \& \ q = (-b - \sqrt{p})/3a \ \& \ r = (f$ при $0 < a$, иначе $e) \ \& \ e \leq x \ \& \ x \leq f \rightarrow ax^3 + bx^2 + cx + d \leq \max(aq^3 + bq^2 + cq + d, ar^3 + br^2 + cr + d))$

Выражения a, b, c, d константные. Два последних антецедента берутся в посылках, причем допускаются строгие неравенства. Второй и третий антецеденты обрабатываются проверочными операторами; первый, четвертый и пятый - выделены указателем "идентификатор". Указатели "подстановка" разрешают обращение в ноль коэффициентов b, c. Свободный член d тоже может быть нулевым. Результирующее выражение обрабатывается вспомогательной задачей на упрощение. Уровень срабатывания равен 3.

13. Логарифм.

$\forall_{abc}(0 < a - 1 \ \& \ b \leq c \rightarrow \log_a b \leq \log_a c)$

Выражение a константное. Первый антецедент обрабатывается проверочным оператором, второй - оператором "верхняяоценка". Уровень срабатывания равен 2.

14. Сумма двух логарифмов.

    $\forall_{abcde}(0 < a - 1 \ \& \ 0 < b \ \& \ d + 2\log_a b \leq e \rightarrow \log_a(b+c) + \log_a(b-c) + d \leq e)$

    Выражение a константное. Первые два антецедента обрабатываются проверочным оператором, третий - оператором "верхняяоценка". Уровень срабатывания равен 1.

15. Целая часть.

    $\forall_{ab}(a \leq b \rightarrow [a] \leq [b])$

    Антецедент обрабатывается оператором "верхняяоценка". Уровень срабатывания равен 1.

    $\forall_{abcdef}(0 \leq a \ \& \ 0 \leq b \ \& \ 0 < c \ \& \ 0 < d \ \& \ 0 < e \ \& \ 0 < f \ \& \ ab/cf \leq p \rightarrow a[bd/ce]e/df \leq p)$

    Последний антецедент обрабатывается оператором "верхняяоценка", остальные антецеденты - проверочными операторами. Любая из переменных может принимать вырожденное единичное значение. Уровень срабатывания равен 3.

16. Обратные тригонометрические функции.

    $\forall_a(\arcsin a \leq \pi/2)$

    $\forall_a(\arccos a \leq \pi)$

    $\forall_a(\mathrm{arctg}\, a \leq \pi/2)$

    $\forall_a(\mathrm{arcctg}\, a \leq \pi)$

    Уровень срабатывания равен 2.

    $\forall_a((\arcsin a)^2 + (arccosa)^2 \leq 5\pi^2/4)$

    Уровень срабатывания равен 1.

17. Мощность множества.

    $\forall_{an}(l(a) = n \rightarrow \mathrm{card}\{;a\} \leq n)$

    Здесь a - конечный набор элементов. Антецедент выделен указателем "идентификатор" и вычисляет длину n этого набора. Уровень срабатывания приема равен 1.

    $\mathrm{card}\emptyset \leq 0$

    Уровень срабатывания равен 1.

    $\forall_{abcmn}(\mathrm{card}a \leq n \ \& \ \mathrm{card}b \leq m \rightarrow \mathrm{card}(a \text{ при } c, \text{ иначе } b) \leq \max(m,n))$

    $\forall_{abmn}(\mathrm{card}a \leq n \ \& \ \mathrm{card}b \leq m \rightarrow \mathrm{card}(a \cup b) \leq m + n)$

    Антецеденты обрабатываются оператором "верхняяоценка". Выражения m, n должный быть натуральными константами. Уровень срабатывания равен 1.

18. Усмотрение результата при наличии комментария "независит".

    $\forall_a(a \leq a)$

    Имеется комментарий (независит ...), переменные которого не входят в a. Тогда выражение a выдается в качестве результата. Уровень срабатывания равен 1.

19. Модуль

(a) Сложение дробей при оценке модуля.

$\forall_{abcde}(d = a/b + c \ \& \ |d| \leq e \rightarrow |a/b + c| \leq e)$

Первый антецедент выделен указателем "идентификатор". Он обрабатывает выражение $a/b + c$ нормализатором "видумножение", выполняющим сложение дробей и предпринимающим попытки разложения числителя и знаменателя на множители. Второй антецедент обращается к оператору "верхняяоценка". Уровень срабатывания равен 3.

(b) Модуль дроби.

$\forall_{abcd}(|a| \leq c \ \& \ d \leq |b| \ \& \ \neg(d = 0) \rightarrow |a/b| \leq c/d)$

Первый антецедент обрабатывается оператором "верхняяоценка", второй - оператором "нижняяоценка". Третий антецедент обрабатывается проверочным оператором. Уровень срабатывания равен 3.

(c) Использование верхней и нижней оценок выражения под модулем.

$\forall_{abc}(b \leq a \ \& \ a \leq c \rightarrow |a| \leq \max(|b|, |c|))$

Первый антецедент обрабатывается оператором "нижняяоценка", второй - оператором "верхняяоценка". Уровень срабатывания равен 4.

(d) Оценка модуля с конечной суммой.

$\forall_{abcfmnpqr}(|q|(\sum_{i=m}^{n} |f(i) + a|)/|r| \leq b \ \& \ |p - (qa(n - m + 1))/r| \leq c \rightarrow |p + q(\sum_{i=m}^{n} f(i))/r| \leq b + c)$

Идентификация начинается с усмотрения в посылках кванторной импликации, имеющей единственную связанную переменную j и содержащей в своем консеквенте выражение $|f(j)+a|$. Затем первый и второй антецеденты обрабатываются операторами "верхняяоценка". Уровень срабатывания приема равен 3.

$\forall_{abcfmnpqr}(|q|(\sum_{i=m}^{n} |a - f(i)|)/|r| \leq b \ \& \ |p + (qa(n - m + 1))/r| \leq c \rightarrow |p + q(\sum_{i=m}^{n} f(i))/r| \leq b + c)$

Аналогично предыдущему, но консеквент кванторной импликации содержит выражение $|a - f(j)|$.

(e) Использование кванторного неравенства из посылок при оценке конечной суммы модулей.

$\forall_{acdmnpqrs}(\forall_j(j - \text{натуральное} \ \& \ c \leq j \rightarrow |f(j) + a| \leq d) \ \& \ p\sum_{i=m}^{c} |f(i) + a|/q \leq r \ \& \ p(n - m + 1)d/q \leq s \rightarrow p\sum_{i=m}^{n} |f(i) + a|/q \leq r + s)$

$\forall_{acdmnpqrs}(\forall_j(j - \text{натуральное} \ \& \ c \leq j \rightarrow |-f(j) + a| \leq d) \ \& \ p\sum_{i=m}^{c} |a - f(i)|/q \leq r \ \& \ p(n - m + 1)d/q \leq s \rightarrow p\sum_{i=m}^{n} |a - f(i)|/q \leq r + s)$

Первый антецедент берется в посылках, причем в его консеквенте допускается строгое неравенство. Второй и третий антецеденты обрабатываются оператором "верхняяоценка". Уровень срабатывания равен 3.

20. Использование монотонности для исключения параметра.

$\forall_{afmnx}(x \in \{m, \ldots, n\} \ \& \ \text{неубывает}(\lambda_x(f(x), x - \text{целое}), \{m, \ldots, n\}) \ \& \ f(n) \leq a \rightarrow f(x) \leq a)$

Первый антецедент берется в посылках, второй - обрабатывается проверочным оператором. x - переменная, входящая в выражение $f(x)$. Переменная f - функциональная, так что последнее выражение - произвольного вида. Введен сравнительно сильный ограничитель трудоемкости. Уровень срабатывания равен 3.

21. Значение функции на элементе множества вычетов.

    $\forall_{fmn}(f(m(\mathrm{mod} n)) \leq \sup(\mathrm{set}_x(\exists_i(i \in \{0, \ldots, n-1\} \ \& \ x = f(i)))))$

    Переменная f - обычная. Имеется комментарий (независит ...), переменные которого не входят в выражения f, n и входят в выражение m. Уровень срабатывания равен 3.

22. Условное выражение.

    $\forall_{abmnP}(a \leq m \ \& \ b \leq n \rightarrow (a \text{ при } P, \text{ иначе } b) \leq \max(m, n))$

    Антецеденты обрабатываются оператором "верхняяоценка". Уровень срабатывания равен 3.

Синтезатор "нижняяоценка"

Оператор аналогичен оператору "верхняяоценка". Антецедент теоремы приема, обращающийся к этому оператору, имеет вид $x \leq a$. Здесь a - некоторое выражение, все переменные которого к моменту обращения уже идентифицированы; x - еще не идентифицированная переменная, которой присваивается после работы синтезатора найденная им нижняя оценка для a. Теорема приема синтезатора имеет своим консеквентом утверждение вида $A \leq B$, где B - оцениваемое выражение, идентифицируемое с поступающим на вход синтезатора выражением, A - верхняя оценка для B, присваиваемая выходной переменной. Приемы синтезатора таковы:

1. Синус и косинус.

    $\forall_a(-1 \leq \sin a)$

    $\forall_a(-1 \leq \cos a)$

    Уровень срабатывания равен 2.

    $\forall_{abx}(a \leq x \ \& \ x \leq b \ \& \ 0 \leq \pi - 2|a| \ \& \ 0 \leq \pi - 2|b| \rightarrow \min(\cos a, \cos b) \leq \cos x)$

    Выражения a, b константные. Первые два антецедента берутся в посылках, причем допускаются строгие неравенства. Третий и четвертый антецеденты обрабатываются проверочным оператором. Уровень срабатывания равен 1.

2. Линейная комбинация синуса и косинуса.

    $\forall_{abc}(-\sqrt{a^2 + c^2} \leq a \sin b + c \cos b)$

    Выражения a, c константные. Уровень срабатывания равен 1.

3. Трехчлен с синусом и косинусом.

    $\forall_{abc}((a + b - \sqrt{(a-b)^2 + c^2})/2 \leq a(\sin x)^2 + b(\cos x)^2 + c \sin x \cos x)$

    Выражения a, b, c константные. Уровень срабатывания равен 1.

4. Плюс.

    $\forall_{abcd}(a \leq b \ \& \ c \leq d \rightarrow a + c \leq b + d)$

    Антецеденты обращаются к оператору "нижняяоценка". Введен указатель "первыйтерм(...)". Уровень срабатывания равен 2.

    $\forall_{abc}(a \leq b \rightarrow c + a \leq c + b)$

Выражение c константное. Антецедент обрабатывается оператором "нижняяоценка". Альтернативные попытки блокируются. Уровень срабатывания приема равен 1.

5. Константа. Аналогично случаю верхней оценки.

6. Минус.

   $\forall_{ab}(b \leq a \rightarrow -a \leq -b)$

   Антецедент обрабатывается оператором "верхняяоценка". Уровень срабатывания равен 2.

   $\forall_{abc}(a + b \leq c \rightarrow -c \leq -a - b)$

   Антецедент обрабатывается оператором "верхняяоценка". Уровень срабатывания равен 1.

7. Умножение.

   $\forall_{abc}(0 < a \ \& \ b \leq c \rightarrow ab \leq ac)$

   Выражение a идентифицируется как отличное от единицы произведение всех константных сомножителей. Первый антецедент обрабатывается проверочным оператором, второй - оператором "нижняяоценка". Альтернативные попытки блокируются. Уровень срабатывания равен 2.

   $\forall_{abcd}(a \leq 0 \ \& \ c \leq a \ \& \ b \leq d \ \& \ 0 \leq d \rightarrow cd \leq ab)$

   Оцениваемое выражение является произведением ровно двух неконстантных множителей a, b. Второй и третий антецеденты обрабатываются, соответственно, операторами "нижняяоценка" и "верхняяоценка". Остальные антецеденты обрабатываются проверочными операторами. Уровень срабатывания равен 3.

   $\forall_{abcd}(0 \leq a \ \& \ 0 \leq c \ \& \ c \leq a \ \& \ d \leq b \rightarrow cd \leq ab)$

   Аналогично предыдущему. Два последних антецедента обрабатываются оператором "нижняяоценка", два первых - проверочным оператором.

   $\forall_{abcd}(a < 0 \ \& \ b < 0 \ \& \ a \leq c \ \& \ b \leq d \ \& \ c < 0 \ \& \ d < 0 \rightarrow cd \leq ab)$

   Третий и четвертый антецеденты обрабатываются оператором "верхняяоценка", остальные антецеденты - проверочными операторами. Уровень срабатывания равен 3.

8. Дробь.

   $\forall_{abc}(0 < c \ \& \ a \leq b \rightarrow a/c \leq b/c)$

   Выражение c константное. Первый антецедент обрабатывается проверочным оператором, второй - оператором "нижняяоценка". Альтернативные попытки блокируются. Уровень срабатывания равен 2.

   $\forall_{abc}(\neg(a = 0) \ \& \ \neg(b = 0) \ \& \ 0 < c \rightarrow 2/c \leq (a^2 + b^2)/c|ab|)$

   Антецеденты обрабатываются проверочными операторами. Уровень срабатывания равен 2.

   $\forall_{abc}(0 < c \ \&c \leq b \ \& \ 0 < a \rightarrow a/b \leq a/c)$

   Выражение a константное. Второй антецедент обрабатывается оператором "верхняяоценка", остальные антецеденты - проверочными операторами. Уровень срабатывания равен 3.

$\forall_{ab}(0 < b \;\&\; 0 \leq a \rightarrow 0 \leq a/b)$

Первый антецедент берется в посылках, второй - обрабатывается проверочным оператором. Уровень срабатывания равен 4.

$\forall_{ab}(0 < b \;\&\; 0 \leq a \rightarrow 0 \leq a/b)$

Антецеденты обрабатываются проверочными операторами. Имеется комментарий (независит ...). Уровень срабатывания равен 4.

$\forall_{abc}(0 < c \;\&\; c \leq b \rightarrow -|a|/c \leq a/b)$

Имеется комментарий (независит ...), переменные которого не входят в a. Второй антецедент обрабатывается оператором "нижняяоценка", первый - проверочным оператором. Уровень срабатывания равен 4.

9. Степень.

   $\forall_{abc}(0 < c \;\&\; b \leq a \;\&\; 0 \leq b \rightarrow b^c \leq a^c)$

   Выражение c константное. Второй антецедент обрабатывается оператором "нижняяоценка", остальные антецеденты - проверочными операторами. Уровень срабатывания равен 1.

   $\forall_{ab}(b - \mathrm{rational} \;\&\; \text{числитель}(b) - \mathrm{even} \rightarrow 0 \leq a^b)$

   Выражение b - константное. Антецеденты обрабатываются проверочными операторами. Уровень срабатывания равен 3.

   $\forall_{abc}(0 < c \;\&\; c - \mathrm{rational} \;\&\; \text{числитель}(c) - \mathrm{even} \;\&\; a \leq b \;\&\; b \leq 0 \rightarrow b^c \leq a^c)$

   Выражение c - константное. Четвертый антецедент обрабатывается оператором "верхняяоценка", остальные - проверочными операторами. Уровень срабатывания равен 1.

   $\forall_{abc}(0 < a - 1 \;\&\; c \leq b \rightarrow a^c \leq b^c)$

   Выражение a константное. Первый антецедент обрабатывается проверочным оператором, второй - оператором "нижняяоценка". Альтернативные попытки блокируются. Уровень срабатывания равен 1.

   $\forall_{abc}(c - \mathrm{rational} \;\&\; \neg(\text{числитель}(c) - \mathrm{even}) \;\&\; \neg(\text{знаменатель}(c) - \mathrm{even}) \;\&\; a \leq b \rightarrow a^c \leq b^c)$

   Первые три антецедента обрабатываются проверочными операторами, последний - оператором "нижняяоценка". Уровень срабатывания равен 1.

10. Использование посылок.

    (a) Неравенство с нулевой частью.
    $\forall_{ab}(0 < a + b \rightarrow -b \leq a)$
    Антецедент берется в посылках, причем допускается нестрогое неравенство. Выражение b константное. В посылках нет аналогичного неравенства, дающего более сильную оценку. Уровень срабатывания равен 3.

    (b) Неравенство для оцениваемого выражения.
    $\forall_{ab}(a < b \rightarrow a \leq b)$
    Антецедент берется в посылках, причем допускается нестрогое неравенство. При наличии комментария (независит ...) его переменные не входят в выражение a, при отсутствии - выражение a константное. Уровень срабатывания равен 3.

(c) Оценка модуля, использующая неравенство с модулем.

$\forall_{abcemnpq}(\neg(m = 0) \;\&\; 0 < e - |am + n| \;\&\; |b| \leq q \;\&\; p \leq |cm - bn| \rightarrow (p - qe)/|m| \leq |ab + c|)$

Второй антецедент берется в посылках, причем допускается нестрогое неравенство. Первый антецедент обрабатывается проверочным оператором, третий - оператором "верхняяоценка", четвертый - оператором "нижняяоценка". Имеется комментарий (независит ...), переменные которого входят в a и не входят в e, m. Уровень срабатывания равен 1.

(d) Транзитивность.

$\forall_{abcx}(x \leq a - c \;\&\; 0 \leq b + c \rightarrow x \leq a + b)$

Второй антецедент берется в посылках, первый - обрабатывается оператором "нижняяоценка". Имеется комментарий (независит ...), причем выражение b неконстантное. Уровень срабатывания равен 3.

(e) Использование неравенства для модуля.

$\forall_{ab}(0 < b - |a| \rightarrow -b \leq a)$

Антецедент берется в посылках, причем допускается нестрогое неравенство. Имеется комментарий (независит ...), переменные которого не входят в b. Уровень срабатывания равен 1.

(f) Использование неравенства для суммы.

$\forall_{ab}(0 < a + b \rightarrow -a \leq b)$

Аналогично предыдущему.

(g) Оценка значений функции натурального аргумента на конечном промежутке.

$\forall_{mn}(n - \text{натуральное} \;\&\; m - \text{целое} \;\&\; 0 \leq m - n \rightarrow \inf(\text{set}_x(\exists_i(i \in \{1, \ldots, m\} \;\&\; x = f(i)))) \leq f(n))$

Первые два антецедента обрабатываются проверочными операторами, последний - берется в посылках. Переменная f - обычная. Имеется комментарий (независит ...), переменные которого входят в n и не входят в m. Уровень срабатывания равен 3.

(h) Элемент конечного отрезка целых чисел.

$\forall_{amn}(a \in \{m, \ldots, n\} \rightarrow m \leq n)$

Антецедент берется в посылках. Имеется комментария (независит ...), переменные которого не входят в m. Уровень срабатывания равен 4.

11. Натуральное.

    $\forall_n(n - \text{натуральное} \rightarrow 1 \leq n)$

    Антецедент берется в посылках, причем n - переменная. Уровень срабатывания равен 4.

12. Оценка для суммы квадратов.

    $\forall_{abcpq}(0 < p \;\&\; 0 < q \;\&\; 0 \leq a \;\&\; 0 \leq b \;\&\; 0 < c \;\&\; c \leq pabq \rightarrow 2\sqrt{2c} \leq pa^2 + pb^2 + q)$

    Выражение p константное. Остаточная сумма q не имеет константных слагаемых. Последний антецедент обрабатывается оператором "нижняяоценка", остальные антецеденты обрабатываются проверочными операторами. Уровень срабатывания равен 1.

13. Целая часть.

    $\forall_{ab}(a \leq b \to [a] \leq [b])$

    Антецедент обрабатывается оператором "нижняяоценка". Уровень срабатывания равен 1.

14. Обратные тригонометрические функции.

    $\forall_{a}(0 \leq a \to 0 \leq \operatorname{arctg} a)$

    Антецедент обрабатывается проверочным оператором. Уровень срабатывания равен 1.

    $\forall_{ab}(b \leq a \ \& \ 0 < b+1 \ \& \ 0 \leq 1-b \to \arcsin b \leq \arcsin a)$

    $\forall_{ab}(a \leq b \ \& \ 0 \leq b+1 \ \& \ 0 < 1-b \to \arccos b \leq \arccos a)$

    Первый антецедент обрабатывается, соответственно, операторами "верхняяоценка" и "нижняяоценка". Второй и третий антецеденты обрабатываются проверочными операторами. Уровень срабатывания равен 1.

    $\forall_{a}(-\pi/2 \leq \arcsin a)$

    $\forall_{a}(0 \leq \arccos a)$

    $\forall_{a}(-\pi/2 \leq \operatorname{arctg} a)$

    $\forall_{a}(0 \leq \operatorname{arcctg} a)$

    Уровень срабатывания равен 2.

15. Логарифм.

    $\forall_{abc}(0 < a-1 \ \& \ b \leq c \to \log_a b \leq \log_a c)$

    Выражение a константное. Первый антецедент обрабатывается проверочным оператором, второй - оператором "нижняяоценка". Уровень срабатывания равен 2.

16. Линейная комбинация взаимно обратных величин.

    $\forall_{abmx}(0 \leq (\operatorname{tg} x)^m \ \& \ 0 < a \ \& \ 0 < b \to 2\sqrt{ab} \leq a(\operatorname{tg} x)^m + b(\operatorname{ctg} x)^m)$

    Выражения a,b константные. Антецеденты обрабатываются проверочными операторами. Уровень срабатывания равен 1.

    $\forall_{ab}(0 \leq a+b+1/(a+b) \to 2-b \leq 1/(a+b)+a)$

    Антецедент обрабатывается проверочным оператором. Уровень срабатывания равен 2.

17. Квадратный трехчлен.

    $\forall_{abcx}(0 < a + c - b^2/4a \leq ax^2 + bx + c)$

    Выражения a, b, c константные. Антецедент обрабатывается проверочным оператором. Уровень срабатывания равен 1.

18. Модуль.

    $\forall_{x}(x - \text{целое} \ \& \ \neg(x = 0) \to 1 \leq |x|)$

    Антецеденты обрабатываются проверочными операторами. Уровень срабатывания равен 1.

$\forall_x(0 \leq |x|)$

Уровень срабатывания равен 2.

19. Усмотрение результата при наличии комментария "независит".

    $\forall_a(a \leq a)$

    Имеется комментарий (независит ...), переменные которого не входят в a. Уровень срабатывания равен 1.

20. Модуль произведения.

    $\forall_{abcd}(c \leq |a| \ \& \ d \leq |b| \to cd \leq |ab|)$

    Антецеденты обрабатываются оператором "нижняяоценка". Уровень срабатывания равен 3.

21. Конечная сумма.

    $\forall_{abfmnpqrs}(\forall_j(j$ – натуральное $\& \ p < j \to q < f(j)) \ \& \ a \leq r(n-p)q/s \ \& \ 0 \leq r \ \& \ 0 < s \ \& \ b \leq r\sum_{i=m}^{p} f(i)/s \to a + b \leq r\sum_{i=m}^{n} f(i)/s)$

    Первый антецедент берется в посылках. Он дает нижнюю оценку q для членов конечной суммы, номер которых больше p. Последний антецедент, обрабатываемый оператором "нижняяоценка", дает нижнюю оценку b для начального отрезка суммы. Второй антецедент тоже обрабатывается оператором "нижняяоценка" - для исключения зависимости от переменных, входящих в коэффициенты r, s и в границы суммирования n, p. Третий и четвертый антецеденты обрабатываются проверочными операторами. Уровень срабатывания равен 3.

22. Частичные сокращения.

    $\forall_{abcdpqx}(p \leq ad/c \ \& \ q \leq bd/cx \to p + q \leq (ax+b)d/cx)$

    Имеется комментарий (независит ...), переменные которого входят в x. Антецеденты обрабатываются оператором "нижняяоценка". Исключение выражения x в первом слагаемом может позволить получить более точную оценку. Уровень срабатывания равен 3.

23. Вещественная часть комплексного числа.

    $\forall_{abz}(|z+a| < b \to -(\mathrm{Re}(a) + b) \leq \mathrm{Re}(z))$

    Сложение и модуль в антецеденте - комплекснозначные. Выражения a, b константные. Антецедент берется в посылках, причем допускается нестрогое неравенство. Уровень срабатывания равен 2.

10.12 Оператор "склейканеравенств"

Нормализатор "склейканеравенств" служит для упрощения дизъюнктивно - конъюнктивных конструкций с неравенствами, возникающих после объединения ответов, соответствующих подслучаям задачи. Обычно он используется в приеме, преобразующем исходное неравенство в дизъюнкцию нескольких конъюнкций. Входящие в конъюнкцию неравенства обрабатываются нормализаторами решения ("уравнменьше", "уравнменьшеилиравно") либо стандартизации ("нормменьше", "нормменьшеилиравно"). После этого вся дизъюнкция передается на обработку нормализатору "склейканеравенств". Он обслуживает одновременно строгие и нестрогие неравенства. Нормализатор "склейканеравенств" не является корневым, т.е. перечисляемые ниже приемы могут быть применены к произвольному внутреннему вхождению в преобразуемое утверждение. Как обычно, кроме основных приемов (для дизъюнкций и конъюнкций неравенств), нормализатор имеет также множество вспомогательных приемов, необходимых для стандартизации возникающих по ходу работы подтермов. Перейдем к перечислению приемов.

1. Конъюнкция двух дизъюнкций с близкими разделителями.

   $\forall_{abcdefgh}((a<b \,\&\, c \;\vee\; b<a \,\&\, d \;\vee\; a=b \,\&\, e) \,\&\, (a<b \,\&\, f \;\vee\; b<a \,\&\, g \;\vee\; a=b \,\&\, h) \leftrightarrow a<b \,\&\, c \,\&\, f \;\vee\; b<a \,\&\, d \,\&\, g \;\vee\; a=b \,\&\, e \,\&\, h)$

   Утверждения c, d, e, f, g, h могут быть вырожденными, и тогда они идентифицируются с константной "истина". Выражения a, b не содержат неизвестных. Роль разделителей рассматриваемых подслучаев играют утверждения $a < b, b < a, a = b$. Несовместность этих утверждений позволяет провести склейку дизъюнкций. Уровень срабатывания равен 1.

   $\forall_{abcdefg}((a \leq b \,\&\, c \;\vee\; b \leq a \,\&\, d) \,\&\, (a<b \,\&\, e \;\vee\; b<a \,\&\, f \;\vee\; a=b \,\&\, g) \leftrightarrow a<b \,\&\, e \,\&\, c \;\vee\; b<a \,\&\, d \,\&\, f \;\vee\; a=b \,\&\, g \,\&\, (c \;\vee\; d))$

   Согласно указателю "подстановка(...)", член $a = b \,\&\, g$ может отсутствовать, и тогда переменная g идентифицируется с константой "ложь". Также допускаются вырожденные c, d, e, f, g, идентифицируемые с константой "истина". Выражения a, b не содержат неизвестных. Уровень срабатывания равен 1.

2. Дизъюнкция двух конъюнкций с близкими разделителями.

   $\forall_{abcdefghpqr}((a=b \,\&\, c \;\vee\; a<b \,\&\, d \;\vee\; b<a \,\&\, e) \,\&\, f \;\vee\; (a=b \,\&\, p \;\vee\; a<b \,\&\, q \vee b<a \,\&\, r) \,\&\, g \leftrightarrow a=b \,\&\, (c \,\&\, f \vee p \,\&\, g) \vee a<b \,\&\, (d \,\&\, f \vee q \,\&\, g) \vee b<a \,\&\, (e \,\&\, f \;\vee\; r \,\&\, g))$

   Допускаются вырожденные выражения $c, d, e, f, g, p, q, r,$, идентифицируемые с константой "истина". Выражения a, b не содержат неизвестных. Уровень срабатывания равен 1.

3. Переход к дизъюнктивной нормальной форме.

   $\forall_{abc}((a \;\vee\; b) \,\&\, c \leftrightarrow a \,\&\, c \;\vee\; b \,\&\, c)$

   Происходит преобразование утверждения к виду дизъюнкции конъюнкций. Это имеет место на раннем этапе работы нормализатора; позднее может быть введен комментарий "нормили", блокирующий данный прием и означающий начало обратного процесса группировки дизъюнктивных членов. Указатель "набор(первыйтерм)" обеспечивает одновременную обработку любого числа дизъюнктивных членов. Каждая конъюнкция обрабатывается нормализатором "склейканеравенств". Уровень срабатывания равен 2.

4. Замена нестрогого неравенства на строгое, если усматривается отрицание равенства.

   $\forall_a(\neg(a=0) \to a \leq 0 \leftrightarrow a < 0)$

   $\forall_a(\neg(a=0) \to 0 \leq a \leftrightarrow 0 < a)$

   Антецедент берется в посылках. Уровень срабатывания равен 3.

5. Дизъюнктивная группировка.

   $\forall_{abc}(a \ \& \ b \ \vee \ a \ \& \ c \leftrightarrow a \ \& \ (b \ \vee \ c))$

   Утверждение a не содержит неизвестных. Вводится комментарий "нормили". Уровень срабатывания равен 4.

   $\forall_{ab}(a \ \vee \ a \ \& \ b \leftrightarrow a)$

   Уровень срабатывания равен 2.

6. Константа "истина".

   $\forall_{a}(a \ \& \ \text{истина} \leftrightarrow a)$

   $\forall_{a}(a \ \vee \ \text{истина} \leftrightarrow \text{истина})$

   Уровень срабатывания равен 1.

7. Константа "ложь".

   $\forall_{a}(a \ \vee \ \text{ложь} \leftrightarrow a)$

   $\forall_{a}(a \ \& \ \text{ложь} \leftrightarrow \text{ложь})$

   Уровень срабатывания равен 1.

8. Вложенные дизъюнкции либо конъюнкции.

   Происходит устранение вложенных дизъюнкций или конъюнкций. Уровень срабатывания равен 1.

9. Перенесение нуля в левую часть, если имеются слагаемые со знаком "минус".

   $\forall_{ab}(a - b < 0 \leftrightarrow 0 < b - a)$

   Прием обеспечивает стандартизацию неравенств, позволяющую усматривать повторные вхождения одного и того же неравенства. Уровень срабатывания равен 2.

10. Учет невырожденности промежутка для неизвестной.

    $\forall_{abcd}(d = (a \ \& \ b < c) \rightarrow a \ \& \ b < x \ \& \ x < c \leftrightarrow d \ \& \ b < x \ \& \ x < c)$

    $\forall_{abcd}(d = (a \ \& \ b \le c) \rightarrow a \ \& \ b \le x \ \& \ x \le c \leftrightarrow d \ \& \ b \le x \ \& \ x \le c)$

    Утверждение a и выражения b, c не содержат неизвестных, x - неизвестная. Из двусторонних неравенств для x косвенным образом вытекает неравенство для b, c. Прием обеспечивает явное вхождение этого неравенства в ответ задачи. Первый из приемов имеет указатели, разрешающие одному из неравенств для x быть нестрогим. Антецедент выделен указателем "идентификатор". Он обрабатывает условие $a \ \& \ b < (\le) c$ на известные параметры нормализатором "склейканеравенств". Преобразуемая конъюнкция - корневая. Проверяется различие утверждений a, d. Для блокировки повторного срабатывания используется комментарий "склейканеравенств". Уровень срабатываания равен 3.

11. Склейка двух односторонних неравенств.

    $\forall_{abcdx}(d = (0 \le b - c) \rightarrow a \ \& \ b < x \ \& \ c < x \leftrightarrow a \ \& \ (d \ \& \ b < x \ \vee \ \neg d \ \& \ c < x))$

    $\forall_{abcdx}(d = (0 \le b - c) \rightarrow a \ \& \ x < b \ \& \ x < c \leftrightarrow a \ \& \ (d \ \& \ x < c \ \vee \ \neg d \ \& \ x < b))$

Выражения b, c не содержат неизвестных, выражение x - содержит. Преобразуемая конъюнкция - корневая. Антецедент выделен указателем "идентификатор". Он обрабатывает неравенство $0 \leq b - c$ нормализатором "стандменьшеилиравно", добавляя к посылкам утверждение a. При идентификации допускается версия нестрогого неравенства для c. Уровень срабатывания равен 3. Добавлены еще два аналогичных приема для случая нестрогих неравенств:

$\forall_{abcdx}(d = (0 \leq b - c) \rightarrow a \ \& \ b \leq x \ \& \ c \leq x \leftrightarrow a \ \& \ (d \ \& \ b \leq x \ \vee \ \neg d \ \& \ c \leq x))$

$\forall_{abcdx}(d = (0 \leq b - c) \rightarrow a \ \& \ x \leq b \ \& \ x \leq c \leftrightarrow a \ \& \ (d \ \& \ x \leq c \ \vee \ \neg d \ \& \ x \leq b))$

12. Подстановка константного значения.

    $\forall_{ab}(a = b \ \& \ f(a) \leftrightarrow a = b \ \& \ f(b))$

    Здесь a - переменная, b - константное выражение. Переменная f - функциональная. Уровень срабатывания равен 1.

13. Стандартизация неравенств для параметров.

    $\forall_{abcd}(d = (a < b) \rightarrow a < b \ \& \ c \leftrightarrow d \ \& \ c)$

    $\forall_{abcd}(d = (a \leq b) \rightarrow a \leq b \ \& \ c \leftrightarrow d \ \& \ c)$

    Неравенство для a, b не содержит неизвестных. Антецедент обрабатывает его, соответственно, нормализатором "стандменьше" либо "стандменьшеилиравно". К посылкам присоединяется утверждение c. Проверяется, что результат d отличается от исходного неравенства и что это неравенство не используется для сопровождения по о.д.з. Уровень срабатывания равен 3.

14. Отрицание неравенства.

    $\forall_{ab}(\neg(a < b) \leftrightarrow b \leq a)$

    $\forall_{ab}(\neg(a \leq b) \leftrightarrow b < a)$

    Уровень срабатывания равен 1.

15. Дизъюнкция неравенства и равенства.

    $\forall_{ab}(a = b \ \vee a \leq b \leftrightarrow a \leq b)$

    Уровень срабатывания равен 1.

16. Избыточность дизъюнктивного члена, имеющего вид неравенства.

    $\forall_{ab}(0 < b - a \rightarrow 0 < a \ \vee \ 0 < b \leftrightarrow 0 < b)$

    $\forall_{ab}(0 < b - a \rightarrow a < 0 \ \vee \ b < 0 \leftrightarrow a < 0)$

    Выражения a, b не содержат неизвестных. Антецедент обрабатывается проверочным оператором. Допускаются случаи нестрогих неравенств. Уровень срабатывания равен 4.

    $\forall_{abcdex}(ac < 0 \ \& \ 0 < (ad - bc)c \rightarrow ax + b \leq 0 \ \vee \ cx + d \leq 0 \ \& \ e \leftrightarrow ax + b \leq 0 \ \vee \ e)$

    $\forall_{abcdex}(0 < ac \ \& \ 0 < (ad - bc)c \rightarrow ax + b \leq 0 \ \vee \ 0 \leq cx + d \ \& \ e \leftrightarrow ax + b \leq 0 \ \vee \ e)$

    $\forall_{abcdx}(ac < 0 \ \& \ 0 < (ad - bc)c \rightarrow ax + b \leq 0 \ \vee \ 0 \leq cx + d \leftrightarrow ax + b \leq 0)$

    $\forall_{abcdx}(ac < 0 \ \& \ (ad - bc)c < 0 \rightarrow ax + b \leq 0 \ \vee \ 0 \leq cx + d \leftrightarrow 0 \leq cx + d)$

    Выражения a, b, c, d константные, x - переменная, не являющаяся неизвестной. Неравенства в преобразуемом утверждении могут быть строгими. Антецеденты обрабатываются проверочными операторами. Уровень срабатывания равен 4.

17. Объединение промежутков для неизвестной.

    $\forall_{abc}((0 \leq a - c) = \text{истина} \rightarrow a < b \ \vee \ c < b \leftrightarrow c < b)$

    Переменная b идентифицируется с неизвестной. В дизъюнкции допускаются нестрогие неравенства. Первый антецедент обрабатывает неравенство оператором "стандменьшеилиравно". Проверка неравенства выполняется здесь усиленными средствами, так как на этапе редактирования ответа сопровождающие утверждения, используемые при быстром усмотрении, могут оказаться уже устраненными. Для срабатывания приема необходимо наличие комментария "стандменьшеилиравно", вводимого внешними процедурами в тех случаях, когда вероятно появление перекрывающихся промежутков значений неизвестной. Уровень срабатывания равен 3.

    $\forall_{abc}((0 \leq b - c) = \text{истина} \rightarrow a < b \ \vee \ a < c \leftrightarrow a < b)$

    Аналогично предыдущему, но неизвестной является a.

18. Избыточность дизъюнктивного члена, имеющего вид равенства.

    $\forall_{afgx}(f(a) \leq g(a) \rightarrow x = a \ \vee \ f(x) \leq g(x) \leftrightarrow f(x) \leq g(x))$

    Выражение a константное, x - переменная, входящая в $f(x)$ либо в $g(x)$. Переменные f, g функциональные. Антецедент обрабатывается проверочным оператором, причем обе его части предварительно преобразуются сквозным нормализатором общей стандартизации "норм". Уровень срабатывания равен 1.

10.13 Приемы символа "максимум"

Выражение "максимум$(a_1 \ldots a_n)$", прорисовываемое формульным редактором в виде $\max(a_1, \ldots, a_n)$, обозначает наибольшее из чисел $a_1, \ldots, a_n$. Справочники "ассоциативно", "коммутативно", "тип" указывают, что эта операция ассоциативна, коммутативна и принимает числовые значения.

Простейшая стандартизация

Простейшая стандартизация обеспечивается приемом, устраняющим вложенные максимумы, и приемом, выполняющим лексикографическое упорядочение операндов максимума. Первый имеет уровень срабатывания 0, второй - уровни срабатывания 0 и 3.

Непосредственное определение максимума

$\forall_a(\max(a, a) = a)$

Заметим, что прием будет срабатывать при любом числе операндов, устраняя повторяющиеся операнды. Уровень срабатывания равен 1.

$\forall_{ab}(0 \leq a - b \rightarrow \max(a, b) = a)$

$\forall_{abc}(0 \leq a - b \rightarrow \max(a + c, b + c) = a + c)$

$\forall_{abc}(0 < c \ \& \ 0 \leq a - b \rightarrow \max(ac, bc) = ac)$

Выражения a, b константные. Антецеденты обрабатываются проверочными операторами. Уровень срабатывания равен 1.

$\forall_{ab}(0 \leq a - b \rightarrow \max(a, b) = a)$

Антецедент обрабатывается проверочным оператором, причем введен средний ограничитель трудоемкости. Уровень срабатывания равен 5.

Символы бесконечности

$\forall_a(\max(a, -\infty) = a)$

$\forall_a(\max(a, \infty) = \infty)$

Уровень срабатывания равен 0.

Стандартизация неравенства

Утверждение о том, что максимум двух чисел меньше третьего числа заменяется парой неравенств для этих чисел:

$\forall_{abcd}(0 \leq c \rightarrow c\max(a, b) + d < 0 \leftrightarrow ac + d < 0 \ \& \ bc + d < 0)$

Допускается случай нестрогого неравенства. Антецедент обрабатывается проверочным оператором. Максимум не должен содержать неизвестных. Уровень срабатывания равен 2.

Вынесение общего множителя

Приемы применяются к каким-либо двум операндам многоместного (вообще говоря) максимума, заменяя их на один новый операнд. Нормализатор "норммаксимум", по мере возможности, исключает максимум в этом операнде.

$\forall_{abcde}(0 \leq a \rightarrow \max(ab/c, ad/e) = a\max(b/c, d/e))$

$\forall_{abcde}(0 < a \rightarrow \max(c/ab, e/ad) = \max(c/b, e/d)/a)$

Антецедент обрабатывается проверочным оператором. Уровень срабатывания приема равен 4.

Решение вспомогательной задачи на описание для определения максимума

Для произвольного операнда максимума предпринимается попытка найти условия, при которых этот операнд не превосходит максимума остальных операндов:

$\forall_{abc}(c = (a \leq b) \rightarrow \max(a, b) = (b \text{ при } c, \text{ иначе } a))$

Здесь a - произвольный операнд максимума, b - максимум оставшихся операндов. Выбирается некоторая общая переменная d выражений a, b, и антецедент обращается к вспомогательной задаче на описание для разрешения неравенства $a \leq b$ относительно d. Условное выражение обрабатывается нормализатором "нормвариант". Уровень срабатывания равен 3.

Разность максимума и минимума

Разность максимума и минимума двух величин заменяется на модуль разности этих величин:

$\forall_{ab}(\max(a, b) - \min(a, b) = |a - b|)$

Уровень срабатывания равен 1.

Усмотрение максимума в условном выражении

Приемы применяются в условии задачи на преобразование:

$\forall_{ab}((b \text{ при } 0 \leq b - a, \text{ иначе } a) = \max(a, b))$

$\forall_{ab}((b \text{ при } 0 < b - a, \text{ иначе } a) = \max(a, b))$

Уровень срабатывания равен 1.

Переход от максимума с неизвестными к условному выражению

Если условие задачи на описание содержит максимум с неизвестными, то он преобразуется к условному выражению:

$\forall_{ab}(\max(a, b) = (a \text{ при } 0 \leq a - b, \text{ иначе } b))$

Выбирается такой максимум, который не содержит внутри себя условного выражения, минимума либо другого максимума, зависящих от неизвестных. Уровень срабатывания равен 2.

Решение простейших неравенств с максимумом

Предусмотрены приемы для неравенств, одна из частей которых представляет собой максимум с неизвестными:

$\forall_{abc}(c \leq \max(a, b) \leftrightarrow c \leq a \ \vee \ c \leq b)$

$\forall_{abc}(\max(a, b) \leq c \leftrightarrow a \leq c \ \& \ b \leq c)$

Приемы допускают случай строгого неравенства. Максимум содержит неизвестную текущей задачи на описание. Уровень срабатывания равен 1.

Попытка рассмотрения отношения сравниваемых выражений

Иногда вместо сравнения двух выражений, расположенных под максимумом, проще сравнить их частное с единицей:

$\forall_{ab}(0 < a \ \& \ 0 < b \ \& \ 0 \leq a/b - 1 \rightarrow \max(a, b) = a)$

Антецеденты обрабатываются проверочными операторами. Попытка применения приема выполняется только в случаях, когда выражение a имеет своим множителем (непосредственным либо по цепочке операций мультипликативного типа - дробей, степеней, минусов) факториал. Уровень срабатывания равен 4.

Разбор случаев

Разбор случаев с целью исключения максимума предусмотрен пока только для максимума двух расстояний:

$\forall_{ABCD}(l(AB) \leq l(CD) \ \vee \ l(CD) \leq l(AB))$

Указатель "контрольвывода(максимум(расстояние(x26 x27)расстояние(x28 x29)))" инициирует прием при обнаружении в посылке задачи на доказательство либо на исследование выражения $\max(l(AB), l(CD))$. Эта посылка не должна содержать численной неизвестной. Выведенная дизъюнкция сопровождается комментарием "разборслучаев". Уровень срабатывания равен 10.

Использование посылок

Если максимумы $\max(a, b), \max(a, c)$ известны и различны, то хотя бы один из них отличен от a, т.е. максимум этих максимумов равен $\max(b, c)$:

$\forall_{abcmn}(\max(a, b) = m \ \& \ \max(a, c) = n \ \& \ \neg(m - n = 0) \to \max(b, c) = \max(m, n))$

Выражения m, n константные. Первые два антецедента берутся в посылках, третий - выделен указателем "программа". Уровень срабатывания равен 2.

$\forall_{abcm}(\max(a, b) = m \ \& \ \max(a, c) = m \to a = m \ \& \ \max(b, c) < m \ \vee \ \max(b, c) = m \ \& \ a \leq m)$

Указатель "контрольвывода(максимум(x2 x3))" инициирует применение приема при обнаружении выражения $\max(b, c)$. Антецеденты берутся в посылках, причем m - константа. Текущая задача на исследование имеет цель "известно". Выведенная дизъюнкция снабжается комментарием "разборслучаев". Уровень срабатывания равен 5.

Усмотрение модуля

Приемы усматривают в максимуме двух отличающихся знаком выражений модуль одного из них:

$\forall_x(\max(x, -x) = |x|)$

$\forall_{ab}(\max(a - b, b - a) = |a - b|)$

Уровень срабатывания равен 1.

Нормализатор "норммаксимум"

Нормализатор имеет прием для непосредственного определения максимума:

$\forall_{ab}(0 \leq a - b \to \max(a, b) = a)$

Антецедент обрабатывается проверочным оператором. Уровень срабатывания приема равен 1. Кроме того, в нормализатор включены приемы для $\pm\infty$, срабатывающие на том же уровне.

10.14 Приемы символа "минимум"

Ограничимся перечислением названий приемов, так как они аналогичны приемам символа "максимум".

1. Простейшая стандартизация - устранение вложенных минимумов и лексикографическое упорядочение операндов.
2. Непосредственное определение минимума.
3. Символы бесконечности.
4. Стандартизация неравенств.
5. Вынесение общего множителя.
6. Решение вспомогательной задачи на описание для определения минимума.

7. Усмотрение минимума в условном выражении
8. Переход от минимума с неизвестными к условному выражению.
9. Решение простейших неравенств с минимумом.
10. Разбор случаев. Кроме приема для минимума двух расстояний, добавлен еще один прием, который тоже ориентирован на задачи типа "исследовать", имеющие цель "известно":

    $\forall_{abx}(x = \min(a, b) \leftrightarrow x = a \ \& \ a \leq b \ \vee \ x = b \ \& \ b < a)$

    Здесь x - неизвестная внешней задачи. Выражения a, b содержат невырожденный числовой атом, встречающийся также в некотором уравнении, не имеющем других неизвестных. Этот прием срабатывает на уровне 10.

11. Исключение минимума под описателем "отображение".

    $\forall_{afA}(\lambda_x(f(\min(a, x)), A(x)) = \lambda_x((f(a) \text{ при } a \leq x, \text{ иначе } f(x)), A(x)))$

    Переменные f, A - функциональные. Указатель "вхождение(x6)" определяет инициализацию приема с обнаружения вхождения выражения $\min(a, b)$ внутри выражения, задающего значения отображения. Замена затрагивает только это вхождение. Уровень срабатывания равен 2.

12. Нормализатор "нормминимум" - аналогично нормализатору "нормаксимум".

10.15 Приемы символа "модуль"

Выражение "модуль(a)", прорисовываемое формульным редактором как $|a|$, обозначает абсолютную величину вещественного числа a. Справочники "арность", "одз", "типданных", "тип" характеризуют простейшие свойства символа "модуль". Заметим, что для модуля комплексных чисел используется другой логический символ - "Модуль", который прорисовывается формульным редактором так же, как и вещественнозначный модуль.

Общая стандартизация выражений

Для стандартизации выражений с модулем созданы следующие приемы:

1. Модуль нуля.

    $|0| = 0$

    Уровень срабатывания равен 0.

2. Минус под модулем.

    $\forall_a(|-a| = |a|)$

    Уровень срабатывания равен 0.

3. Двойной модуль.

    $\forall_a(||a|| = |a|)$

    Уровень срабатывания равен 0.

4. Произведение модулей.

   Произведение модулей преобразуется к виду модуля произведения:

   $\forall_{ab}(|a||b| = |ab|)$

   Уровень срабатывания равен 0.

   $\forall_{bcde}(\neg(\text{знаменатель}(d) - \text{even})\ \&\ \neg(\text{знаменатель}(e) - \text{even})\ \&\ d - \text{rational}\ \&\ e - \text{rational} \rightarrow |b|^e|c|^d = |b^e c^d|)$

   Для ускоренной идентификации введен фильтр "постпозиция(фикс(0 1 2)фикс(0 1 1))", отсекающий симметричные случаи. d может обращаться в единицу, e - невырожденное. Уровень срабатывания равен 1.

   $\forall_{abcde}(\neg(\text{знаменатель}(d) - \text{even})\ \&\ \neg(\text{знаменатель}(e) - \text{even})\ \&\ d - \text{rational}\ \&\ e - \text{rational} \rightarrow |c|^a|b^e c^d| = |b|^e|c|^{a+d})$

   Показатели степени a, d, e могут обращаться в единицу. Уровень срабатывания равен 0.

5. Модуль дроби.

   Если из посылок удается усмотреть пропорциональность для двух модулей, коэффициенты которой не содержат неизвестных, то содержащий неизвестные модуль дроби заменяется отношением данных коэффициентов:

   $\forall_{abcd}(\neg(d = 0)\ \&\ d|a| = c|b| \rightarrow |a/b| = c/d)$

   Первый антецедент обрабатывается проверочным оператором, второй - берется в посылках. Текущая задача имеет тип "исследовать". Хотя бы одно из выражений a, b содержит неизвестные; выражения c, d неизвестных не содержат. Уровень срабатывания равен 1.

   $\forall_{abcd}(|a| = b\ \&\ |c| = d \rightarrow |a/c| = b/d)$

   Оба антецедента берутся в посылках текущей задачи на исследование. Выражения a, c содержат неизвестные, выражения b, d - не содержат. Уровень срабатывания равен 7.

6. Устранение модуля при помощи оператора "усмменьшеилиравно".

   $\forall_a(a - \text{число}\ \&\ 0 \leq a \rightarrow |a| = a)$

   Антецеденты обрабатываются проверочными операторами, причем для обработки второго из них введен средний ограничитель трудоемкости. При вычислении предела либо при выдаче ответа, инициируемой целью "ответ", прием блокируется. Он не применяется также к посылкам задачи на доказательство либо на исследование, содержащим символ "площадь". Указатель "заменавхождений" определяет одновременную замену всех вхождений в задачу данного модуля, для которых контекст содержит обоснование корректности замены. Уровни срабатывания равны 0,1 и 3.

   $\forall_a(a \leq 0 \rightarrow |a| = -a)$

   Аналогично предыдущему, но ограничитель трудоемкости несколько более сильный, а уровни срабатывания равны 0 и 3.

   Кроме указанных приемов, на тех же теоремах созданы еще два приема, используемые в задачах на исследование, имеющих цель "контроль". Такая цель вводится при рассмотрении подслучая, который может оказаться противоречивым. Выражение a не должно содержать неизвестных. Приемы имеют сильный ограничитель трудоемкости. Уровень срабатывания их равен 2.

7. Устранение модуля при помощи оператора "провменьшеилиравно".

   $\forall_a(a - \text{число} \ \& \ 0 \leq a \rightarrow |a| = a)$

   $\forall_a(a - \text{число} \ \& \ a \leq 0 \rightarrow |a| = a)$

   Приемы применяются к выражению, входящему в условие задачи на доказательство. Второй антецедент обрабатывается усиленным оператором усмотрения нестрогих неравенств "провменьшеилиравно". Уровень срабатывания равен 3.

8. Свертка условного выражения в модуль.

   $\forall_a(a - \text{число} \rightarrow (-a \text{ при } a \leq 0, \text{ иначе } a) = |a|)$

   $\forall_a(a - \text{число} \rightarrow (a \text{ при } 0 \leq a, \text{ иначе } -a) = |a|)$

   $\forall_a(a - \text{число} \rightarrow (a \text{ при } 0 < a, \text{ иначе } -a) = |a|)$

   $\forall_a(a - \text{число} \rightarrow (-a \text{ при } a < 0, \text{ иначе } a) = |a|)$

   Указатель "извлечениеварианта" существенно усиливает прием. Вместо того чтобы идентифицировать альтернативы рассматриваемого условного выражения с $a, -a$, прием предпринимает попытку представить их в виде $T(a), T(-a)$ для некоторого общего "шаблона" T. После этого условное выражение заменяется на $T(|a|)$. Антецедент обрабатывается проверочным оператором. Имеется слабый ограничитель трудоемкости. Если имеется комментарий (нормвариант A), запрещающий разбор случаев при рассмотрении условного выражения A, содержащего преобразуемое приемом условное выражение, то прием блокируется. Уровень срабатывания равен 0.

9. Деление на модуль дроби.

   $\forall_{abdeg}(d/g|b/a|^e = d|a/b|^e/g)$

   Уровень срабатывания равен 0.

   $\forall_{abdeg}(\neg(\text{знаменатель}(e) - \text{even} \ \& \ e - \text{rational} \rightarrow d/g(-|b/a|)^e = d(-1)^e|a/b|^e/g)$

   Прием применяется для преобразования условия задачи на преобразование, имеющей цель "упростить". Если в задаче выделены неизвестные, то они не должны входить в d. Антецеденты обрабатываются проверочными операторами. Уровень срабатывания равен 4.

10. Вынесение из-под модуля множителей при определении их знака.

    $\forall_{ab}(0 \leq b \rightarrow |ab| = b|a|)$

    $\forall_{ab}(b \leq 0 \rightarrow |ab| = -b|a|)$

    Антецедент обрабатывается проверочным оператором. Уровни срабатывания равны 1 и 3.

11. Вынесение из-под модуля множителей числителя либо знаменателя дроби при определении их знака.

    $\forall_{abc}(0 \leq b \rightarrow |bc/a| = b|c/a|)$

$\forall_{abc}(0 \leq b \rightarrow |a/bc| = |a/c|/b)$

$\forall_{abc}(b < 0 \rightarrow |a/bc| = -|a/c|/b)$

$\forall_{abc}(b \leq 0 \rightarrow |bc/a| = -b|c/a|)$

$\forall_{ab}(0 \leq b \rightarrow |b/a| = b/|a|)$

Антецедент обрабатывается проверочным оператором. При наличии цели "ответ", указывающей на необходимость немедленной выдачи ответа, прием блокируется. Уровни срабатывания равны 1 и 3.

12. Вынесение из-под модуля суммы дробей общего множителя числителей.

    $\forall_{abcdm}(0 \leq m \rightarrow |am/b + cm/d| = m|a/b + c/d|)$

    Требуется, чтобы общий множитель m имел вхождение невырожденного (отличного от переменной и неконстантного) числового атома. Антецедент обрабатывается проверочным оператором. Уровень срабатывания приема равен 3.

13. Группировка модулей - множителей числителя и знаменателя дроби.

    $\forall_{abdefg}(\neg(\text{знаменатель}(f) - \text{even})\ \&\ \neg(\text{знаменатель}(g) - \text{even})\ \&\ f - \text{rational}\ \&\ g - \text{rational} \rightarrow d|e|^f/a|b|^g = d|e^f/b^g|/a)$

    Антецеденты обрабатываются проверочными операторами. Выражения a, d, f, g могут вырождаться в единицу. Уровень срабатывания равен 1.

14. Переход к сигнуму.

    $\forall_{abc}(|a|b/ac = \mathrm{sg}(a)b/c)$

    На этой теореме созданы два приема. Первый - применяется, если рассматриваемая дробь является операндом одной из операций "косинус", "модуль", "арккосинус". Он имеет уровень срабатывания 1. Второй прием применяется без ограничений и имеет уровень срабатывания 6.

    $\forall_{abc}(ab/|a|c = \mathrm{sg}(a)b/c)$

    Аналогично предыдущему: на уровне 1 срабатывает версия, применяемая, если дробь является операндом одной из операций "косинус", "модуль", "арккосинус", на уровне 6 - версия для общего случая. Кроме того, созданы еще две версии приема, срабатывающие на уровне 3 в задачах на преобразование. Одна из них требует наличие цели "сигнум", другая - наличие цели "нормИнтеграл" (упрощение подынтегрального выражения при формальном интегрировании).

    $\forall_{abc}(a|b/ac| = |b/c|\mathrm{sg}(a))$

    $\forall_{acdn}(\neg(n - \text{even})\ \&\ n - \text{натуральное} \rightarrow ca^n/d|a|^n = c\mathrm{sg}(a)/d)$

    Показатель степени n во втором приеме невырожденный. Уровень срабатывания равен 6.

15. Сокращение на рациональную степень модуля.

    $\forall_{abcde}(a - \text{rational}\ \&\ \text{числитель}(a) - \text{even} \rightarrow c^a d/|c|^b e = |c|^{a-b}d/e)$

    Указатель "дробь(фикс(0 1)фикс(0 2))" разрешает перестановку местами числителя и знаменателя. Уровень срабатывания равен 0.

    $\forall_{abcmn}(ba^m/c|a|^n = ba^{m-n}\mathrm{sg}(a)/c)$

Переменная n идентифицируется с целочисленной нечетной константной (возможно, вырождающейся в единицу). Прием применяется в задачах на описание, имеющих цель "связка", т.е. фактически - только при решении дифференциальных уравнений. Уровень срабатывания равен 0.

$\forall_{abcdef}(\text{числитель}(a) - \text{even} \ \& \ a - \text{rational} \rightarrow dc^a f^a/e|cf|^b = d|cf|^{a-b}/e)$

Уровень срабатывания равен 0.

16. Модуль степени.

    $\forall_{ab}(b - \text{rational} \ \& \ \neg(\text{знаменатель}(b) - \text{even}) \rightarrow |a^b| = |a|^b)$

    Уровень срабатывания равен 1.

17. Четная степень модуля.

    $\forall_{ap}(p - \text{rational} \ \& \ \text{числитель}(p) - \text{even} \rightarrow |a|^p = a^p)$

    Антецеденты здесь и в предыдущих приемах обрабатываются проверочными операторами. Уровень срабатывания равен 0.

    $\forall_{abc}(b = 2c \ \& \ c - \text{целое} \rightarrow |a|^b = a^b)$

    Первый антецедент выделен указателем "идентификатор", второй - обрабатывается проверочным оператором. Уровень срабатывания равен 0.

18. Модуль разности. Выполняется изменение знаков для перехода к лексикографически меньшему выражению:

    $\forall_{abc}(a = b - c \rightarrow |c - b| = |a|)$

    Антецедент выделен указателем "идентификатор". Он меняет знаки всех слагаемых модуля, причем хотя бы одно слагаемое должно было иметь знак "минус". Результат изменения знаков и стандартного их переупорядочения оказывается лексикографически предшествующим исходной разности. Уровень срабатывания равен 1.

19. Отбрасывание модуля в произведении суммы и разности.

    $\forall_{abc}((a - |b|c)(a + |b|c) = (a - bc)(a + bc))$

    Уровень срабатывания равен 1.

20. Сумма двух выражений, отличающихся только знаком перед модулем. В такой сумме модуль можно отбросить из соображений симметрии:

    $\forall_{abcft}(t = f(-a|b|) \rightarrow f(a|b|) + t + c = f(ab) + f(-ab) + c)$

    Переменная f - функциональная. Указатель "вхождение(x6)" определяет идентификацию подвыражения $a|b|$ в произвольной внутренней точке слагаемого, идентифицируемого с $f(a|b|)$. Коэффициент a может обращаться в единицу. После того, как слагаемое $f(a|b|)$ идентифицировано, оно дает шаблон для построения выражений вида $f(X)$: вхождение $a|b|$ будет заменяться на терм X. Антецедент, выделенный указателем "идентификатор", находит результат t обработки нормализаторами общей стандартизации выражения $f(-a|b|)$. Проверяется, что слагаемые выражения t включаются в остаточные слагаемые суммы, после чего идентифицируется остаточная часть суммы c. Введен средний ограничитель трудоемкости. Для очень громоздких выражений прием блокируется. Уровень срабатывания равен 5.

21. Сокращение с занесением под знак модуля.

    $\forall_{abcd}(0 \leq bc \rightarrow b|ac|/cd = |ab|/d)$

    $\forall_{abc}(0 \leq bc \rightarrow |ab/c| = b|a|/c)$

    $\forall_{abc}(0 \leq bc \rightarrow bc|a/c| = |ab|)$

    Антецедент обрабатывается проверочным оператором. Уровень срабатывания равен 4.

22. Степень модуля произведения.

    $\forall_{abc}(\neg(\text{знаменатель}(b) - \text{even})\ \&\ \neg(\text{числитель}(b) - \text{even})\ \&\ b - \text{rational} \rightarrow |ac^{1/b}|^b = |ca^b|)$

    Антецеденты обрабатываются проверочными операторами. Уровень срабатывания равен 0.

23. Степень минус единицы.

    $\forall_{ap}(p - \text{rational}\ \&\ \neg(\text{знаменатель}(p) - \text{even}) \rightarrow |a(-1)^p| = |a|)$

    Уровень срабатывания равен 0.

24. Сложение дробей под модулем.

    $\forall_{abcp}(0 < c \rightarrow |ap + bp/c| = |p(ac + b)|/c)$

    Выражение p неконстантное. Модуль не расположен под квантором либо описателем, либо внутри неравенства. Если решается задача на описание, то она не имеет цели (связка ...). Таким образом отсекаются случаи решения дифференциальных уравнений. Антецедент обрабатывается проверочным оператором. Уровень срабатывания равен 3.

Общая стандартизация утверждений

1. Равенство модуля нулю.

   $\forall_a(|a| = 0 \leftrightarrow a = 0)$

   Уровень срабатывания равен 0.

2. Свертка конъюнкции двух неравенств в неравенство с модулем.

   $\forall_{ab}(a - b \leq 0\ \&\ 0 \leq a + b \leftrightarrow -b + |a| \leq 0)$

   $\forall_{ab}(a + b < 0\ \&\ 0 < a - b \leftrightarrow b + |a| < 0)$

   $\forall_{ab}(0 < a + b\ \&\ 0 < b - a \leftrightarrow -b + |a| < 0)$

   $\forall_{ab}(a + b < 0\ \&\ a - b < 0 \leftrightarrow a + |b| < 0)$

   $\forall_{ab}(a + b \leq 0\ \&\ b - a \leq 0 \leftrightarrow b + |a| \leq 0)$

   $\forall_{ab}(0 \leq a + b\ \&\ 0 \leq a - b \leftrightarrow -a + |b| \leq 0)$

   Приемы имеют заголовок "второйтерм", т.е. применяются к двум неравенствам, входящим в одну конъюнкцию. Текущая задача не должна иметь тип "доказать". Если конъюнкция расположена в условии задачи, то выражения a, b не должны содержать неизвестных. Уровень срабатывания равен 2.

3. Свертка дизъюнкции двух неравенств в неравенство с модулем.

   $\forall_{ab}(a+b<0 \ \vee \ 0<a-b \leftrightarrow 0<-b+|a|)$

   $\forall_{ab}(a-b\leq 0 \ \vee \ 0\leq a+b \leftrightarrow 0\leq b+|a|)$

   $\forall_{ab}(0<a+b \ \vee \ 0<b-a \leftrightarrow 0<b+|a|)$

   $\forall_{ab}(0\leq a+b \ \vee \ 0\leq a-b \leftrightarrow 0\leq a+|b|)$

   $\forall_{ab}(a+b\leq 0 \ \vee \ b-a\leq 0 \leftrightarrow 0<-b+|a|)$

   $\forall_{ab}(a+b<0 \ \vee \ a-b<0 \leftrightarrow 0<-a+|b|)$

   Приемы имеют заголовок "второйтерм". Если преобразуемая дизъюнкция расположена в условии задачи на описание, то выражения a,b не содержат неизвестных. Уровень срабатывания равен 2.

4. Свертка дизъюнкции двух равенств в равенство с модулем.

   $\forall_{ab}(a=b \ \& \ 0\leq b \ \vee \ b=-a \ \& \ a\leq 0 \leftrightarrow b=|a|)$

   $\forall_{ab}(a - \text{число} \rightarrow a=b \ \& \ 0\leq b \ \vee \ a=-b \ \& \ b\leq 0 \leftrightarrow a=|b|)$

   $\forall_{ab}((a=b \ \vee \ a=-b) \ \& \ 0\leq a \leftrightarrow a=|b|)$

   $\forall_{ab}(a+b=0 \ \& \ 0\leq a \ \vee \ b-a=0 \ \& \ b\leq 0 \leftrightarrow b+|a|=0)$

   $\forall_{ab}(a+b=0 \ \& \ 0\leq a \ \vee \ b-a=0 \ \& \ a\leq 0 \leftrightarrow b+|a|=0)$

   $\forall_{ab}((a+b=0 \ \vee \ b-a=0) \ \& \ b\leq 0 \leftrightarrow b+|a|=0)$

   $\forall_{ab}(a+b=0 \ \& \ b\leq 0 \ \vee \ b-a=0 \ \& \ a\leq 0 \leftrightarrow b+|a|=0)$

   $\forall_{ab}(a+b=0 \ \& \ 0\leq a \ \vee \ a-b=0 \ \& \ a\leq 0 \leftrightarrow b+|a|=0)$

   $\forall_{ab}((a+b=0 \ \vee \ a-b=0) \ \& \ b\leq 0 \leftrightarrow b+|a|=0)$

   Если дизъюнкция входит в условие задачи на описание, то выражение под модулем не имеет неизвестных. Уровень срабатывания равен 2.

   $\forall_{ax}(x=a \ \& \ 0<a \ \vee \ x=-a \ \& \ a<0 \leftrightarrow x=|a| \ \& \ \neg(a=0))$

   Дизъюнкция входит в посылку задачи на исследование. Выражение a не содержит неизвестных. Уровень срабатывания равен 0.

5. Развертка неравенства с модулем в дизъюнкцию двух неравенств.

   $\forall_{ab}(0<a+|b| \leftrightarrow 0<a+b \ \vee \ 0<a-b)$

   Предполагается, что неравенство расположено в области действия описателя "класс", связывающего какие-либо параметры выражения b. Допускается случай нестрогого неравенства. Уровень срабатывания равен 2.

6. Неравенство с произведением выражения на его модуль.

   $\forall_{ab}(0\leq ab|b| \leftrightarrow 0\leq ab)$

   $\forall_{ab}(0<ab|b| \leftrightarrow 0<ab)$

   Указатель "дробь" позволяет применять прием с перестановкой частей неравенства. Уровень срабатывания равен 0.

7. Равенство модулей сумм, отличающихся знаком.

   $\forall_{ab}(|a-b| = |b-a|)$

   Указатель "эквивалентно" сообщает компилятору, что прием выполняет не замену $|a-b|$ на $|b-a|$, а замену равенства $|a-b| = |b-a|$ на константу "истина". Идентификация выполняется в следующем порядке. Сначала находится слагаемое $-a$ суммы под вторым модулем. Это позволяет идентифицировать b как сумму остальных слагаемых. Указатель "нормзнака(x2 минус плюс)" определяет идентификацию выражения $-b$ в сумме под первым модулем как сумму слагаемых b с измененными знаками. Уровень срабатывания равен 0.

8. Положительность модуля.

   $\forall_a(0 < |a| \leftrightarrow \neg(a = 0))$

   Замена выполняется даже в случаях, когда неравенство используется для сопровождения по о.д.з. Уровень срабатывания равен 4.

9. Равенство выражения своему модулю.

   $\forall_a(a = |a| \leftrightarrow 0 \leq a)$

   Уровень срабатывания равен 1.

   $\forall_{ab}(\neg(b = 0) \rightarrow ba - b|a| = 0 \leftrightarrow 0 \leq a)$

   Антецедент обрабатывается проверочным оператором. Указатель "заменазнака(минус x2)" обеспечивает возможность идентификации с измененными знаками слагаемых (тогда минус относится к b). Допускается вырожденное единичное значение b. Уровень срабатывания равен 1.

10. Равенство нулю суммы выражения и его модуля.

    $\forall_a(a + |a| = 0 \leftrightarrow a \leq 0)$

    $\forall_a(-a = |a| \leftrightarrow a \leq 0)$

    Уровень срабатывания равен 0.

    $\forall_a(a = -|a| \leftrightarrow a \leq 0)$

    Уровень срабатывания равен 1.

11. Неположительность модуля.

    $\forall_a(|a| \leq 0 \leftrightarrow a = 0)$

    Уровень срабатывания равен 0.

12. Исключение модуля из дизъюнкции равенств.

    $\forall_{abc}(a = b|c| \ \vee \ a = -b|c| \leftrightarrow a = bc \ \vee \ a = -bc)$

    Уровень срабатывания равен 0.

13. Усмотрение равенства величин из равенства их модулей и неотрицательности произведения.

    $\forall_{ab}(0 \leq ab \rightarrow |a| - |b| = 0 \leftrightarrow a - b = 0)$

    $\forall_{ab}(ab \leq 0 \rightarrow |a| - |b| = 0 \leftrightarrow a + b = 0)$

    Антецедент обрабатывается проверочным оператором, причем введен сравнительно сильный ограничитель трудоемкости. Уровень срабатывания равен 1.

14. Разбор случаев по знаку переменной, встречающейся в консеквенте кванторной импликации под модулем. Если данная переменная - связанная, то кванторная импликация разбивается на две независимых кванторных импликации, для неотрицательных и отрицательных значений переменной:

    $\forall_{PQ}(\forall_{xy}(x$ – число & $P(x,y) \to Q(x,y)) \leftrightarrow \forall_{xy}(x$ – число & $0 \leq x$ & $P(x,y) \to Q(x,y))$ & $\forall_{xy}(x$ – число & $x < 0$ $P(x,y) \to Q(x,y)))$

    Проверяется наличие в консеквенте вхождения $|x|$, где x ранее идентифицировано по антецеденту "x – число". Переменные P, Q - функциональные. Указатель "кортежпеременных(x24)" разрешает идентифицировать y с произвольным (в том числе пустым) набором переменных. Преобразуемая кванторная импликация должна находиться в условии задачи на описание. Уровень срабатывания равен 2.

15. Исключение модуля из соотношения пропорциональности.

    $\forall_{abcde}(\neg(a=0)$ & $0 < bcde \to ab/c = |a|d/e \leftrightarrow 0 < a$ & $b/c = d/e)$

    Антецеденты обрабатываются проверочными операторами. Переменные b, c, d, e могут идентифицироваться с единицей. Решаемая задача должна иметь тип "исследовать" и цель "известно". Уровень срабатывания равен 2.

16. Сокращение на модуль.

    $\forall_{abcd}(\neg(b=0) \to 0 \leq a|b| + |bc|d \leftrightarrow 0 \leq a + |c|d)$

    Допускаются перестановка частей неравенства и рассмотрение строгого неравенства. Переменные a, c, d могут идентифицироваться с единицей. Уровень срабатывания равен 0.

    $\forall_{abc}(ab^2 = c|b| \leftrightarrow b = 0 \ \vee \ a|b| = c)$

    Решается задача на исследование, имеющая цель "известно". Выражение b содержит неизвестные, выражения a, c - не содержат. Уровень срабатывания равен 1.

17. Исключение модуля в уравнении. Прием выполняет исключение модуля в равенстве специального вида, являющемся посылкой задачи на исследование, имеющей цель "известно":

    $\forall_{abcx}(0 \leq x$ & $0 \leq a$ & $0 \leq c$ & $0 \leq c - a$ & $b < 0 \to |ax+b| = cx \leftrightarrow -ax - b = cx$ & $ax + b \leq 0)$

    Выражение x содержит неизвестные, выражения a, b, c - не содержат. Антецеденты обрабатываются проверочными операторами. Прием полезен в тех случаях, когда x имеет невырожденные числовые атомы, т.е. не происходит немедленного обращения к вспомогательной задаче для разрешения уравнения относительно x. Уровень срабатывания равен 2.

18. Исключение модуля в посылке.

    $\forall_{ab}(0 < b - a \to |a| = b \leftrightarrow a + b = 0)$

    $\forall_{ab}(0 < a + b$ & b – число $\to |a| = b \leftrightarrow a = b)$

    Равенство является посылкой задачи на доказательство либо задачи на исследование, имеющей цель "известно". Антецеденты обрабатываются проверочными операторами. Имеется сильный ограничитель трудоемкости. Уровни срабатывания равны 3 и 5.

Усмотрение неотрицательности модуля

$\forall_a(0 \leq |a|)$

Прием заменяет неравенство на константу "истина". Уровень срабатывания равен 1.

Эквивалентные преобразования условий с неизвестными

1. Простейшее уравнение.

   $\forall_{ab}(b - \text{число} \rightarrow b = |a| \leftrightarrow (a = b \ \vee \ a = -b) \ \& \ 0 \leq b)$

   Равенство входит в условие задачи на описание, причем допустимые надтермы имеют один из заголовков "и", "или", "не". Выражение a содержит неизвестные, выражение b - не содержит. Задача не имеет цели (связка ...). Так как обычно уравнения с модулем решаются с помощью разбора случаев, прием ориентирован лишь на остаточные случаи, где разбор случаев был заблокирован. Уровень срабатывания равен 5.

2. Решение неравенств с модулем.

   $\forall_{ab}(|a| < b \leftrightarrow a < b \ \& \ -b < a)$

   $\forall_{ab}(|a| \leq b \leftrightarrow a \leq b \ \& \ -b \leq a)$

   $\forall_{ab}(b \leq |a| \leftrightarrow a \leq -b \ \vee \ b \leq a)$

   $\forall_{ab}(b < |a| \leftrightarrow a < -b \ \vee \ b < a)$

   Неравенство находится в условии задачи на описание; допускаются только надтермы с заголовками "и", "или", "не", "существует". Выражение a содержит неизвестные, выражение b - не содержит. В двух последних приемах выражение b ненулевое, в двух первых - задача имеет цель "полный" и не имеет цели "пример". Уровень срабатывания равен 1. Создана также версия второго приема, срсбатывающая на уровне 5 и относящаяся к задачам на описание, имеющим цель "пример" либо не имеющим цели "полный".

   $\forall_a(0 < |a| \leftrightarrow \neg(a = 0))$

   Аналогично предыдущему, но уровень срабатывания равен 0. Неравенство не должно располагаться в условии, выделенном комментарием "разборслучаев".

3. Замена дизъюнкции неравенств с неизвестным выражением на неравенство для модуля.

   $\forall_{ab}(b < a \ \vee \ b < -a \leftrightarrow b < |a|)$

   $\forall_{ab}(b \leq a \ \vee \ b \leq -a \leftrightarrow b \leq |a|)$

   Дизъюнкция входит в условие задачи на описание. Выражение b содержит неизвестные, выражение a - не содержит. Указатель "дизъюнктоперанд" блокирует анализ неявных вхождений дизъюнкции. Уровень срабатывания равен 1 (второй прием при наличии цели "пример" срабатывает на уровне 3).

4. Замена двух неравенств с неизвестным выражением на неравенство для модуля.

   $\forall_{ab}(a < b \ \& \ -a < b \leftrightarrow |a| < b)$

   $\forall_{ab}(a \leq b \ \& \ -a \leq b \leftrightarrow |a| \leq b)$

   Неравенства суть условия задачи на описание. Выражение a не содержит неизвестных, выражение b - содержит. Уровни срабатывания равны 1,3 и 5.

5. Свертка дизъюнкции равенств в равенство модулю.

   $\forall_{ax}(x -$ число $\& \ 0 \leq x \rightarrow x = a \ \vee \ x = -a \leftrightarrow x = |a|)$

   Дизъюнкция является условием задачи на описание. Переменная x идентифицируется с неизвестной, выражение a неизвестных не содержит. Указатель "дизъюнктоперанд" блокирует попытки усмотрения дизъюнкции в неявных случаях. Второй антецедент берется в посылках. Уровень срабатывания приема равен 0.

6. Упрощение соотношения для суммы квадратов величин с использованием линейного соотношения для модуля.

   $\forall_{abcdexy}(\neg(a = 0) \ \& \ a|x| + by = 0 \rightarrow cx^2 + dy^2 = e \leftrightarrow (cb^2 + da^2)y^2 = ea^2)$

   Решается задача на исследование. Второй антецедент берется в посылках, первый - обрабатывается проверочным оператором. Выражения x, y содержат неизвестные, выражения a, b, c, d, e - не содержат. Указатель "группировки(x24 фикс(2 1 2))" определяет идентификацию члена by путем группировки всех слагаемых, делящихся на ранее идентифицированное y. Уровень срабатывания равен 3.

7. Использование равенства с модулем.

   $\forall_{abx}(0 \leq a \ \& \ |x| = c \ \& \ 0 \leq c - a \rightarrow |a + x| = b \leftrightarrow 0 \leq b \ \& \ x = -b - a)$

   Прием применяется в задачах на исследование, имеющих невырожденные числовые атомы. Выражения a, b не содержат неизвестных; невырожденные числовые атомы выражения x связаны соотношениями с численными неизвестными внешней задачи на описание. Второй антецедент берется в посылках, первый и третий - обрабатываются проверочными операторами. Уровень срабатывания равен 4.

   $\forall_{abcd}(0 \leq a \ \& \ |b| = d \ \& \ c < 0 \rightarrow ab = c \leftrightarrow ad = -c)$

   Решается задача на исследование, имеющая цель "известно". Второй антецедент берется в посылках, первый и третий - обрабатываются проверочным оператором. Выражение b имеет невырожденный числовой атом, выражение d - не имеет. Уровень срабатывания равен 2.

Разбор случаев по знаку неизвестного выражения под модулем

Для исключения неизвестного модуля в задачах на описание обычно применяется следующий прием разбора случаев:

$\forall_a(a \leq 0 \ \vee \ 0 < a)$

Прием имеет заголовок "выводусловия". Указатель "контрольвывода(модуль(x1))" определяет его инициализацию при усмотрении вхождения выражения $|a|$. Проверяется, что это вхождение - в условие задачи на описание и что a содержит неизвестные. Проверяются также следующие требования:

1. Текущее условие не имеет более короткого неизвестного модуля либо сигнума.
2. Задача не имеет своим условием дизъюнкцию с неизвестными.
3. Либо текущее условие - уравнение, либо задача не имеет уравнений.

4. Рассматриваемое вхождение модуля не расположено в области действия кванторов и описателей по переменным, входящим в модуль.
5. Задача имеет цель "полный" и не имеет цели "пример".
6. Рассматриваемое вхождение модуля не расположено в суммируемом или интегрируемом выражении.
7. Не имеет места этап редактирования найденного ответа задачи.
8. Максимальный уровень задачи не менее 4.
9. Отсутствуют специальные цели "связка ..." (решение дифференциальных уравнений) и "замещение".

Выведенная дизъюнкция снабжается комментарием "разборслучаев". Уровень срабатывания равен 2.

$\forall_{ab}(|a| = b \to a = b \ \vee \ a = -b)$

Прием имеет заголовок "вывод". Решается задача на исследование, имеющая цель "известно". Антецедент берется в посылках, причем a - содержащий неизвестные числовой атом, b - не содержит неизвестных. Дизъюнкция сопровождается комментарием "разборслучаев". Уровень срабатывания равен 9. На этой же теореме создан аналогичный прием, в котором a - просто выражение с неизвестными, не обязательно числовой атом. Его уровень срабатывания равен 10.

Разбор случаев по знаку выражения под модулем в задачах на преобразование

Как и в предыдущем разделе, прием основан на теореме $\forall_a(a \leq 0 \ \vee \ 0 < a)$. Он имеет заголовок "вывод". Указатель "контрольвывода(модуль(x1))" инициализирует попытку применения при обнаружении выражения $|a|$ в условии задачи на преобразование. Проверяется выполнение следующих требований:

1. Преобразуемое выражение имеет хотя бы одно вхождение параметра a вне рассматриваемого вхождения модуля.
2. Модуль не расположен внутри квантора, описателя либо логарифма.
3. Если преобразуемое выражение имеет единственное вхождение символа "модуль", то оно не является сомножителем данного выражения.
4. Преобразуемое выражение не чрезмерно велико: если оно содержит тригонометрическую операцию либо логарифм, то длина его не более 100, иначе - не более 150.
5. Задача не имеет специальных целей "класс", "значениеинтеграла".

Выводимая посылка сопровождается комментарием "разборслучаев". Уровень срабатывания равен 7. Создана еще одна версия приема - для задач с целью "нормИнтеграл", выполняющих преобразование подынтегрального выражения при формальном интегрировании. Модуль должен содержать переменную интегрирования и располагаться под радикалом. При этом a должно отличаться от переменной. Уровень срабатывания этой версии равен 5.

Разбор случаев по знаку известного выражения под модулем

Прием применяется при редактировании ответа неравенств. Теорема и инициализация приема - прежние. Заголовок - "выводусловия". Для срабатывания необходимо выполнение следующих требований:

1. Модуль неконстантный. Он расположен в условии задачи на описание и не содержит неизвестных.
2. Имеет место этап редактирования найденного ответа задачи.
3. Либо задача имеет неравенство с неизвестными, либо все ее условия с неизвестными имеют вид "число(x)".
4. Рассматриваемое вхождение модуля не расположено в области действия кванторов и описателей по переменным, входящим в модуль.
5. Текущее условие задачи не имеет вхождения более длинного модуля.
6. Задача не имеет специальных целей "связка ...", "нормобласть", "редуцирование".

Уровень срабатывания равен 3.

Разбор случаев по знаку выражения под модулем при решении дифференциального уравнения

При решении дифференциальных уравнений понадобились несколько дополнительных приемов разбора случаев по знаку выражения под модулем - как содержащего, так и не содержащего неизвестных.

$\forall_{abcd}(a|b| + c = d \to 0 \leq b \ \vee \ b < 0)$

Антецедент представляет собой условие задачи на описание, имеющей цель (связка $x_1 \dots x_n$). Такая цель перечисляет варьируемые переменные $x_1, \dots, x_n$ решаемых функциональных уравнений. В условии должна встречаться производная, содержащая неизвестные. Каждое слагаемое выражений c, d должно делиться на b либо на модуль b. Задача не должна иметь дизъюнктивного условия. Выводимая дизъюнкция снабжается комментарием "разборслучаев". Созданы две версии приема. Первая имеет заголовок "выводусловия" и требует, чтобы b содержало неизвестные - она срабатывает на уровне 1. Вторая имеет заголовок "вывод" и требует, чтобы b не содержало неизвестных, но содержало варьируемые параметры. Она срабатывает на уровне 3.

$\forall_{abcdexy}((b|a| + c)/d + e \cdot dy(x)/dx = 0 \to a = 0 \ \vee \ 0 < a \ \vee \ a < 0)$

Прием имеет заголовок "выводусловия". Антецедент идентифицируется с условием задачи на описание, имеющей цель (связка ...). Переменная y - неизвестная. Выражение a содержит неизвестные, выражения b, d - не содержат. Каждое слагаемое выражения c представляет собой произведение a либо $|a|$ на известный коэффициент, причем имеется хоты бы одно такое слагаемое с a. Задача не имеет дизъюнктивного условия. Уровень срабатывания равен 4.

Вывод следствий

1. Вывод следствия из равенства модулей.

   $\forall_{abc}(|a| = |b| \ \& \ 0 < ac \ \& \ bc < 0 \to a = -b)$

   $\forall_{abc}(|a| = |b| \ \& \ 0 < ac \ \& \ 0 < bc \to a = b)$

   $\forall_{abc}(|a| = |b| \ \& \ ac < 0 \ \& \ bc < 0 \to a = b)$

   Все антецеденты берутся в посылках, причем указатель "равно(1)" позволяет идентифицировать равенство модулей из двух равенств их третьему выражению. У первого приема a - переменная, не входящая в b, остальные два приема применяются без ограничений. Уровень срабатывания равен 4.

2. Модуль разности неизвестных величин известен.

   $\forall_{abcdef}(\neg(a = 0) \ \& \ |b - d| = e \ \& \ ab + c = ad + f \to |(f - c)/a| = e)$

   Прием имеет заголовок "вывод" и применяется в задачах на исследование. Второй и третий антецеденты берутся в посылках, первый - обрабатывается проверочным оператором. Выражения b, d содержат неизвестные, выражение e - не содержит. Кроме того, хотя бы одно из выражений c, f содержит неизвестные. Уровень срабатывания равен 4.

3. Вывод следствий из неравенств для модуля.

   $\forall_{ab}(|a| < b \to a < b)$

   $\forall_{ab}(|a| < b \to -a < b)$

   Приемы применяются в задачах на исследование, имеющих цель "длялюбого". Такие задачи возникают при рассмотрении кванторных условий внешней задачи на описание. Выражение a должно содержать неизвестные. Допускается случай нестрогих неравенств. Уровень срабатывания равен 3.

Группировка посылок, приводящая к появлению модуля

$\forall_{ab}((a + b = 0 \ \vee \ a - b = 0) \ \& \ b \leq 0 \leftrightarrow b + |a| = 0)$

$\forall_{ab}((a + b = 0 \ \vee \ b - a = 0) \ \& \ b \leq 0 \leftrightarrow b + |a| = 0)$

$\forall_{ab}((a = b \ \vee \ a = -b) \ \& \ 0 \leq a \leftrightarrow a = |b|)$

Приемы имеют заголовок "заменатермов(второйтерм)". Они заменяют две посылки задачи (дизъюнкцию и неравенство) на равенство с модулем. Уровень срабатывания равен 2.

Группировка условий, приводящая к появлению модуля

$\forall_{ab}(a - b \leq 0 \ \& \ 0 \leq a + b \leftrightarrow -b + |a| \leq 0)$

$\forall_{ab}(a + b < 0 \ \& \ 0 < a - b \leftrightarrow b + |a| < 0)$

$\forall_{ab}(a + b \leq 0 \ \& \ b - a \leq 0 \leftrightarrow b + |a| \leq 0)$

$\forall_{ab}(0 \leq a + b \ \& \ 0 \leq a - b \leftrightarrow -a + |b| \leq 0)$

$\forall_{ab}(a + b < 0 \ \& \ a - b < 0 \leftrightarrow a + |b| < 0)$

$\forall_{ab}(0 < a + b \ \& \ 0 < b - a \leftrightarrow -b + |a| < 0)$

Приемы имеют заголовок "заменаусловия(второйтерм)". Они применяются к двум неравенствам в условиях задачи на описание на этапе редактирования ответа. Оба неравенства не содержат неизвестных и не используются для сопровождения по о.д.з. Ненулевая часть неравенств не должна иметь вида $m + nx$, где m, n - константы, x - переменная. Уровни срабатывания равны 2, 4 и 6.

$\forall_{ab}((a + b = 0 \ \vee \ a - b = 0) \ \& \ b \leq 0 \leftrightarrow b + |a| = 0)$

$\forall_{ab}((a + b = 0 \ \vee \ b - a = 0) \ \& \ b \leq 0 \leftrightarrow b + |a| = 0)$

$\forall_{ab}((a = b \ \vee \ a = -b) \ \& \ 0 \leq a \leftrightarrow a = |b|)$

Приемы имеют заголовок "заменаусловия(второйтерм)". Они применяются к двум условиям задачи на описание. Выражение под модулем не должно содержать неизвестных. Уровни срабатывания равны 0, 2 и 4.

Нормализатор общей стандартизации "нормодуль"

Приемы нормализатора аналогичны приведенным выше приемам общей стандартизации выражений с модулем, поэтому ограничимся перечислением их названий:

1. Модуль нуля.
2. Минус под модулем.
3. Двойной модуль.
4. Устранение модуля при помощи оператора "усмменьшеилиравно".
5. Вынесение из-под модуля множителей при определении их знака.
6. Вынесение из-под модуля множителей числителя либо знаменателя дроби при определении их знака.
7. Модуль степени.
8. Символы бесконечности.

   $|-\infty| = \infty$

   $\forall_A(|(-\infty \text{ при } A, \text{ иначе } \infty)| = \infty)$

   $\forall_A(|(\infty \text{ при } A, \text{ иначе } -\infty)| = \infty)$
9. Условное выражение.

   $\forall_{Ab}(|(b \text{ при } A, \text{ иначе } -b)| = |b|)$

   $\forall_{Ab}(|(-b \text{ при } A, \text{ иначе } b)| = |b|)$

Нормализатор стандартизации модулей с неизвестными "уравнмодуль"

Нормализатор имеет единственный прием, передающий для дальнейшей обработки процедурой "нормуравн" выражение под модулем:

$\forall_{ab}(b = a \rightarrow |a| = |b|)$

Антецедент выделен указателем "идентификатор". Он присваивает переменной b результат обработки выражения a нормализатором "нормуравн". Заменяющий терм $|b|$ обрабатывается нормализатором "нормодуль".

10.16 Приемы символа "сигнум"

Выражение "сигнум(a)", прорисовываемое формульным редактором в виде $\mathrm{sg}(a)$, равно -1 для отрицательных значений a, не определено в нуле и равно 1 для положительных значений a.

Общая стандартизация выражений

1. Попытка исключения сигнума путем установления знака.

   $\forall_a(a \le 0 \to \mathrm{sg}(a) = -1)$

   $\forall_a(0 \le a \to \mathrm{sg}(a) = 1)$

   Антецедент обрабатывается проверочным оператором. Уровень срабатывания равен 1.

   $\forall_a(\mathrm{sg}(a) \le 0 \to \mathrm{sg}(a) = -1)$

   $\forall_a(0 \le \mathrm{sg}(a) \to \mathrm{sg}(a) = 1)$

   Антецедент берется в посылках. Уровень срабатывания равен 3.

2. Отбрасывание множителя с известным знаком.

   $\forall_{ab}(b < 0 \to \mathrm{sg}(ab) = -\mathrm{sg}(a))$

   $\forall_{ab}(0 < b \to \mathrm{sg}(ab) = \mathrm{sg}(a))$

   Антецедент обрабатывается проверочным оператором. Уровень срабатывания равен 2.

3. Дробь под сигнумом.

   $\forall_{ab}(\mathrm{sg}(b/a) = \mathrm{sg}(ab))$

   Уровень срабатывания равен 0.

4. Минус под сигнумом.

   $\forall_a(\mathrm{sg}(-a) = -\mathrm{sg}(a))$

   Уровень срабатывания равен 0.

5. Степень под сигнумом.

   $\forall_{ab}(a - \mathrm{rational}\ \&\ \neg(\text{знаменатель}(a) - \mathrm{even})\ \&\ \neg(\text{числитель}(a) - \mathrm{even}) \to \mathrm{sg}(b^a) = \mathrm{sg}(b))$

   Антецеденты обрабатываются проверочными операторами. Уровень срабатывания равен 1.

6. Двойной сигнум.

   $\forall_{ac}(\mathrm{sg}(a\mathrm{sg}(c)) = \mathrm{sg}(ac))$

   Уровень срабатывания равен 0.

7. Произведение модуля на сигнум.

   $\forall_{an}(|a|^n \mathrm{sg}(a) = a^n)$

   Переменная n идентифицируется с нечетной натуральной константой. Уровень срабатывания равен 0.

$\forall_{ab}(|a/b|\mathrm{sg}(ab) = a/b)$

Уровень срабатывания равен 0.

8. Степень сигнума.

   $\forall_{ab}(a - \mathrm{rational}\ \&\ \text{числитель}(a) - \mathrm{even} \to (\mathrm{sg}(b))^a = 1)$

   Антецеденты обрабатываются проверочными операторами. Уровень срабатывания равен 0.

   $\forall_{ab}(a{-}\mathrm{rational}\ \&\ \neg(\text{знаменатель}(a){-}\mathrm{even})\ \&\ \neg(\text{числитель}(a){-}\mathrm{even}) \to (\mathrm{sg}(b))^a = \mathrm{sg}(b))$

   $\forall_{abc}(\neg(\text{знаменатель}(c){-}\mathrm{even})\ \&\ \text{числитель}(b){-}\mathrm{even}\ \&\ b{-}\mathrm{rational}\ \&\ c{-}\mathrm{rational} \to (\mathrm{sg}(a))^{b+c} = (\mathrm{sg}(a))^c)$

   Антецеденты обрабатываются проверочными операторами. Уровень срабатывания равен 1.

9. Деление сигнума на модуль.

   $\forall_{abcd}(d\mathrm{sg}(ab)/c|a/b| = db/ac)$

   Указатель "дробь(фикс(0 1)фикс(0 2))" разрешает перестановку числителя и знаменателя. Уровень срабатывания равен 0.

   $\forall_{abc}(a\mathrm{sg}(b)/c|b| = a/bc)$

   Допускается перестановка числителя и знаменателя. Уровень срабатывания равен 1.

   $\forall_{abc}(\mathrm{sg}(a)|b/ac| = |b/c|/a)$

   Уровень срабатывания равен 2.

10. Группировка под общий сигнум с исключением модуля.

    $\forall_{abc}(a|b| + c\mathrm{sg}(b) = (ab + c)\mathrm{sg}(b))$

    Выражение не расположено под описателем "класс". Уровень срабатывания равен 1.

11. Занесение минуса под сигнум.

    $\forall_{abcd}(-a\mathrm{sg}(b - c)/d = a\mathrm{sg}(c - b)/d)$

    Уровень срабатывания равен 1.

12. Произведение сигнумов.

    $\forall_{ab}(\mathrm{sg}(a)\mathrm{sg}(b) = \mathrm{sg}(ab))$

    Уровень срабатывания равен 1.

13. Сигнум гиперболического синуса.

    $\forall_{ac}(\mathrm{sg}(a\,\mathrm{sh}\,c) = \mathrm{sg}(ac))$

    Уровень срабатывания равен 0.

14. Сигнум синуса и косинуса.

    $\forall_a(\mathrm{sg}(\sin a) = (-1)^{[a/\pi]})$

    $\forall_a(\mathrm{sg}(\cos a) = (-1)^{[a/\pi+1/2]})$

    Во вспомогательных задачах на преобразование, используемых нормализаторами приемов, данные замены блокируются. Уровень срабатывания равен 2.

15. Знак определителя второго порядка.

    $\forall_{abcdefpq}(de + bf = 0 \,\&\, \mathrm{sg}(ae - cf) = p \rightarrow \mathrm{sg}((ab + cd)q) = \mathrm{sg}(q(be - df))p)$

    Прием преобразует посылку задачи на исследование либо на доказательство. Первый антецедент берется в посылках. Он устанавливает линейную зависимость между неконстантными выражениями b, d, линейная комбинация которых расположена под сигнумом. Второй антецедент, выделенный указателем "идентификатор", вычисляет с помощью нормализаторов общей стандартизации знак p определителя двух линейных соотношений. Заменяющий терм содержит под сигнумом новую линейную комбинацию выражений b, d. Проверяется, что параметры исходной линейной комбинации $ab+cd$ не накрывались параметрами никакой имеющейся в посылках линейной зависимости для b, d. Уровень срабатывания равен 5.

16. Приведение подобных членов с сигнумами сумм, отличающихся знаком.

    $\forall_{abcd}(a\mathrm{sg}(b - c) + d\mathrm{sg}(c - b) = (a - d)\mathrm{sg}(b - c))$

    Сумма не расположена под описателем "класс", связывающим какие-либо переменные коэффициентов a, d. Уровень срабатывания равен 2.

17. Переход к модулю.

    $\forall_a(a\mathrm{sg}(a) = |a|)$

    Прием применяется, если каждое вхождение сигнума a в преобразуемый терм является сомножителем произведения, содержащего также a. Уровень срабатывания равен 1.

    $\forall_{ab}(0 \le ab \rightarrow b\mathrm{sg}(a) = |b|)$

    $\forall_{ab}(bc \le 0 \rightarrow b\mathrm{sg}(c) = -|b|)$

    Антецедент обрабатывается проверочным оператором. При решении дифференциальных уравнений прием блокируется. Уровень срабатывания равен 4.

    $\forall_{abc}(a\mathrm{sg}(b)/bc = a/|b|c)$

    К условию задачи на преобразование прием применяется только при наличии цели "длина", определяющей завершающее редактирование терма с целью его сокращенной перезаписи. Уровень срабатывания равен 2.

18. Модуль сигнума.

    $\forall_a(\neg(a = 0) \rightarrow |\mathrm{sg}(a)| = 1)$

    Антецедент обрабатывается проверочным оператором. Уровень срабатывания равен 0.

19. Свертка условного выражения с использованием сигнума.

    $\forall_{ab}(\neg(a=0)\ \&\ b - \text{число} \rightarrow (b \text{ при } 0 \leq a,\ \text{иначе } -b) = b\mathrm{sg}(a))$

    $\forall_{ab}(\neg(a=0)\ \&\ b - \text{число} \rightarrow (b \text{ при } 0 < a,\ \text{иначе } -b) = b\mathrm{sg}(a))$

    $\forall_{ab}(\neg(a=0)\ \&\ b - \text{число} \rightarrow (b \text{ при } a < 0,\ \text{иначе } -b) = -b\mathrm{sg}(a))$

    Во втором приеме допускается нестрогое неравенство. Должен отсутствовать комментарий (нормвариант A), у которого A содержит преобразуемое условное выражение. Уровень срабатывания равен 0.

20. Произведение сигнума на приращение функции.

    $\forall_{abcf}(c = a - b\ \&\ f(x) = -g(x) \rightarrow \mathrm{sg}(c)(f(\max(a,b)) + g(\min(a,b))) = f(a) - f(b))$

    Указатель "контекст(позиция(x4 корень)вид(x4 максимум(x1 x2)))" определяет идентификацию выражений a, b как операндов некоторого максимума, входящего в преобразуемый терм. Переменные f, g - функциональные. Указатели "новаргумент(x6 набор(x1 x2)фикс)", "новаргумент(x7 набор(x1 x2)фикс)" обеспечивают проверку того, что переменные выражений a, b входят в f только внутри выражений $\max(a,b)$, а в g - только внутри выражений $\min(a,b)$. Первый и второй антецеденты выделены указателями "идентификатор". Они проверяют, соответственно, что выражение c под сигнумом равно разности a, b и что выражения $f(x), g(x)$ отличаются только знаком. Уровень срабатывания равен 1.

21. Перенесение сигнума из знаменателя в числитель.

    $\forall_{abc}(a/b\mathrm{sg}(c) = a\mathrm{sg}(c)/b)$

    Уровень срабатывания равен 0.

22. Произведение сигнума на сумму с сигнумом.

    $\forall_{abc}((a + b\mathrm{sg}(c))\mathrm{sg}(c) = a\mathrm{sg}(c) + b)$

    Выражение a не должно представлять собой сумму. Уровень срабатывания равен 1.

23. Сокращение дроби с сигнумами.

    $\forall_{abcpq}((a\mathrm{sg}(b) + c)p/(a + c\mathrm{sg}(b))q = p\mathrm{sg}(b)/q)$

    Уровень срабатывания равен 1.

Общая стандартизация утверждений

1. Равенство нулю произведения с сигнумом.

   $\forall_{ab}(a\mathrm{sg}(b) = 0 \leftrightarrow a = 0)$

   Уровень срабатывания равен 0.

2. Неравенства с сигнумом.

   $\forall_{a}(0 < \mathrm{sg}(a) + 1 \leftrightarrow 0 \leq a)$

   $\forall_{a}(0 < 1 - \mathrm{sg}(a) \leftrightarrow a \leq 0)$

   $\forall_{a}(\mathrm{sg}(a) \leq 1)$

   $\forall_{a}(1 \leq \mathrm{sg}(a) \leftrightarrow 0 < a)$

$\forall_a(0 < \mathrm{sg}(a) \leftrightarrow 0 < a)$

$\forall_a(0 \leq \mathrm{sg}(a) \leftrightarrow 0 < a)$

Все приемы имеют заголовок "второйтерм". В последних двух приемах допускается перестановка частей неравенства. Уровни срабатывания равны 0.

3. Исключение сигнума из дизъюнкции равенств.

$$\forall_{abcdx}(x = \frac{a + \mathrm{sg}(b)c}{d} \ \vee \ x = \frac{a - \mathrm{sg}(b)c}{d} \leftrightarrow x = \frac{a + c}{d} \ \vee \ x = \frac{a - c}{d})$$

$$\forall_{abcdx}(x = -\frac{a + \mathrm{sg}(b)c}{d} \ \vee \ x = \frac{\mathrm{sg}(b)c - a}{d} \leftrightarrow x = -\frac{a + c}{d} \ \vee \ x = \frac{c - a}{d})$$

Уровень срабатывания равен 2.

4. Равенство сигнума единице либо минус единице.

$\forall_a(\mathrm{sg}(a) = 1 \leftrightarrow 0 < a)$

$\forall_a(\mathrm{sg}(a) = -1 \leftrightarrow a < 0)$

Уровень срабатывания равен 1.

$\forall_a(1 + \mathrm{sg}(a) = 0 \leftrightarrow a < 0)$

$\forall_b(-\mathrm{sg}(b) = 1 \leftrightarrow b < 0)$

Уровень срабатывания равен 2.

5. Отличие сигнума от нуля.

$\forall_a(a - \text{число} \ \& \ \neg(a = 0) \rightarrow \neg(\mathrm{sg}(a) = 0))$

Антецеденты обрабатываются проверочными операторами. Уровень срабатывания равен 1.

Вынесение сигнума из-под нечетных функций

$\forall_{abc}(\mathrm{arctg}(\mathrm{sg}(a)b/c) = \mathrm{sg}(a)\,\mathrm{arctg}(b/c))$

$\forall_{abc}(\arcsin(\mathrm{sg}(a)b/c) = \mathrm{sg}(a)\arcsin(b/c))$

Уровень срабатывания равен 0.

Разбор случаев по знаку сигнума с неизвестными

$\forall_a(a \leq 0 \ \vee \ 0 < a)$

Прием имеет заголовок "выводусловия". Указатель "контрольвывода(сигнум(x1))" определяет инициализацию его применения при усмотрении в условии задачи на описание выражения $\mathrm{sg}(a)$, содержащего неизвестные. Дополнительно проверяется выполнение следующих требований:

1. Максимальный уровень задачи не менее 4.
2. Текущее условие не имеет более длинного модуля либо сигнума, содержащего неизвестные.
3. Отсутствует дизъюнктивное условие задачи, содержащее неизвестные.

4. Либо текущее условие представляет собой уравнение, либо задача не имеет уравнений.
5. Задача не имеет цели (связка ...).

Выведенное условие сопровождается комментарием "разборслучаев". Уровень срабатывания равен 2.

Определение сигнума из уравнения

$\forall_{abcd}(a\mathrm{sg}(c)/b = d \ \& \ \neg(a = 0) \ \& \ \neg(b = 0) \ \& \ \mathrm{sg}(abd) = e \to \mathrm{sg}(c) = e)$

Прием имеет заголовок "вывод" и применяется в задачаха на доказательство либо на исследование. Первый антецедент берется в посылках, второй и третий - обрабатываются проверочными операторами. Четвертый антецедент выделен указателем "идентификатор". Он обрабатывает выражение $\mathrm{sg}(abd)$ вспомогательной задачей на преобразование, решаемой с максимальным уровнем 4. Проверяется, что результат e представляет собой константу. Уровень срабатывания равен 4.

Нормализатор общей стандартизации "нормсигнум"

Нормализатор содержит двойники следующих приемов общей стандартизации:

1. Попытка исключения сигнума путем установления знака.
2. Отбрасывание множителя с известным знаком.
3. Дробь под сигнумом.
4. Минус под сигнумом.
5. Степень под сигнумом.
6. Двойной сигнум.

10.17 Приемы символа "Сигнум"

Кроме сигнума, не определенного в нуле, введена еще одна аналогичная функция, определенная уже на всей числовой прямой. Она принимает значение 1 в неотрицательных точках и 0 - в отрицательных. Эта функция, обозначаемая "Сигнум(a)", прорисовывается формульным редактором как $\mathrm{Sg}(a)$. Для "Сигнум"а созданы всего два приема сканирования задачи:

$\forall_a(a < 0 \to \mathrm{Sg}(a) = -1)$

$\forall_a(0 \leq a \to \mathrm{Sg}(a) = 1)$

Антецедент обрабатывается проверочным оператором. Уровень срабатывания равен 1.

Кроме того, создан нормализатор общей стандартизации "нормСигнум", в котором эти два приема продублированы.

10.18 Приемы символа "логарифм"

Выражение "логарифм(a b)", прорисовываемое формульным редактором в виде $\log_a b$, обозначает логарифм вещественного числа b по положительному отличному от единицы основанию a. Если основание равно числу e, формульный редактор прорисовывает логарифм в виде $\ln a$. Специального обозначения для десятичного логарифма не предусмотрено.

Общая стандартизация выражений с логарифмами

1. Логарифм единицы.

   $\forall_a(\log_a 1 = 0)$

   Уровень срабатывания равен 0.

2. Число под логарифмом равно основанию логарифма.

   $\forall_a(\log_a a = 1)$

   Уровень срабатывания равен 0.

3. Логарифм произведения.

   $\forall_{abc}(0 \leq a \ \& \ 0 \leq b \rightarrow \log_c(ab) = \log_c a + \log_c b)$

   Антецеденты обрабатываются проверочными операторами. Указатель "заменавхождений" определяет одновременную замену всех таких вхождений в задачу данного логарифма, контекст которых содержит все использованные проверочными операторами утверждения. Прием блокируется при завершающем редактировании ответа задач на преобразование либо на описание. Он блокируется также при рассмотрении условия $\log_c(ab) = d$ задачи на описание, где левая часть содержит неизвестные, а правая - нет. Уровень срабатывания приема равен 1.

   $\forall_{abc}(a \leq 0 \ \& \ b \leq 0 \rightarrow \log_c(ab) = \log_c(-a) + \log_c(-b))$

   Прием блокируется при редактировании ответа задачи на преобразование, а также в условиях задач на описание, если c содержит неизвестные, а a, b - не содержат. В остальном - аналогично предыдущему.

4. Логарифм дроби.

   $\forall_{abc}(0 \leq a \ \& \ 0 \leq b \rightarrow \log_c(a/b) = \log_c a - \log_c b)$

   $\forall_{abc}(a \leq 0 \ \& \ b \leq 0 \rightarrow \log_c(a/b) = \log_c(-a) - \log_c(-b))$

   Антецеденты обрабатываются проверочными операторами. Как и выше, замена выполняется сразу для всех вхождений логарифма в задачу. Дополнительно проверяется выполнение следующих требований:

   (a) Не имеет места этап редактирования ответа задачи на преобразование.

   (b) Либо текущий терм задачи имеет вхождение другого логарифма, не расположенное внутри преобразуемого логарифма, но достижимое из него путем переходов через символы алгебраических операций, равенства и неравенства, либо решается задача на преобразование выражения для разложения его в ряд Тейлора.

(c) Если логарифм находится в условии задачи на описание либо в посылке задачи на исследование и содержит неизвестные, то должно найтись вхождение неизвестной в тот же терм задачи, не расположенное внутри $\log_c(a/b)$ либо внутри $\log_{a/b} c$.

Уровень срабатывания равен 1.

$\forall_{abc}(\log_a(b/c) = \log_a|b| - \log_a|c|)$

$\forall_{abc}(\log_a|b/c| = \log_a|b| - \log_a|c|)$

Для применения преобразования необходимо выполнение следующих требований:

(a) Логарифм входит в условие задачи на преобразование, не решаемой для упрощения производной, либо в условие задачи на описание. В обоих случаях не имеет места этап завершающего редактирования ответа.

(b) Если решается задача на описание, то хотя бы одно из выражений b, c содержит неизвестные.

(c) Текущий терм задачи имеет вхождение другого логарифма $\log_A B$, не расположенное внутри преобразуемого вхождения, причем выражение B имеет b своим обобщенным множителем (допустимы переходы через операции "умножение", "дробь", "минус" и через основания степеней).

Кроме того, первый прием блокируется при решении дифференциальных уравнений. Уровень срабатывания равен 3.

$\forall_{ab}(0 < ab \mathbin{\&} \neg(a/b - 1 = 0) \rightarrow \log_{a/b}(b/a) = -1)$

$\forall_{cd}(\log_{|c/d|}|d/c| = -1)$

$\forall_{ab}(\log_{-b/a}(-a/b) = -1)$

Уровень срабатывания равен 0.

$\forall_{ab}(\log_a(1/b) = -\log_a b)$

Выражение b не имеет заголовка "дробь". Уровень срабатывания равен 0.

$\forall_{abcd}(0 \leq a \mathbin{\&} 0 \leq bc \rightarrow \log_d(ab/c) = \log_d a + \log_d(b/c))$

$\forall_{abcd}(0 \leq a \mathbin{\&} 0 \leq bc \rightarrow \log_d(b/ac) = \log_d(b/c) - \log_d a)$

Выражение a идентифицируется с отдельным невырожденным сомножителем, остаточное произведение тоже невырожденное. Антецеденты обрабатываются проверочными операторами. Не имеет места этап завершающего редактирования ответа задачи на преобразование. Замена предпринимается сразу для всех вхождений логарифма в задачу. Уровень срабатывания равен 2.

$\forall_{abc}(\ln((a\mathrm{sg}(b) + c)/b) = \ln(a + c\mathrm{sg}b) - \ln|b|)$

Прием применяется при преобразовании подынтегрального выражения при формальном интегрировании. Уровень срабатывания равен 4.

5. Логарифм степени.

$\forall_{abc}(0 \leq a \rightarrow \log_b(a^c) = c\log_b a)$

$\forall_{abc}(a \leq 0 \mathbin{\&} b - \mathrm{rational} \mathbin{\&} \text{числитель}(b) - \mathrm{even} \rightarrow \log_c(a^b) = b\log_c(-a))$

$\forall_{abc}(a \le 0 \;\&\; c - \text{rational} \;\&\; \neg(\text{числитель}(c) - \text{even}) \;\&\; \neg(\text{знаменатель}(c) - \text{even}) \to \log_b(-a^c) = c\log_b(-a))$

Антецеденты обрабатываются проверочными операторами. Заменяются все вхождения логарифма в задачу. Уровень срабатывания равен 1.

$\forall_{abc}(\log_a(b^c) = c\log_a|b|)$

Логарифм расположен в условии задачи на преобразование либо на описание, причем не имеет места этап редактирования ответа. В случае задачи на описание выражение b должно содержать неизвестные. Либо задача имеет цель "связка ...", либо логарифм расположен под описателем "отображение", либо существует другое вхождение логарифма $\log_A B$, не расположенное внутри преобразуемого вхождения, и b является обобщенным множителем выражения B. Заменяются все вхождения логарифма в задачу. Уровень срабатывания приема равен 2.

$\forall_{abcd}(\log_a(b^c/d^c) = c\log_a|b/d|)$

Логарифм находится в условии задачи на преобразование под описателем "отображение". Показатель степени c - невырожденный. Чтобы идентифицировать его как общий множитель показателей степеней сомножителей числителя и знаменателя, введены указатели "общаястепень(общаястепень x3 степень(x2 x3)степень(x4 x3))". Уровень срабатывания приема равен 2.

$\forall_{ab}(\log_a(a^b) = b)$

$\forall_{bc}(\text{числитель}(c) - \text{even} \;\&\; c - \text{rational} \to \log_{|b|}(b^c) = c)$

Антецеденты обрабатываются проверочными операторами. Уровень срабатывания равен 0.

$\forall_{abmnkp}(pm - kn = 0 \to \log_{a^n b^m}(a^p b^k) = p/n)$

Антецедент выделен указателем "идентификатор". Он обрабатывает левую часть равенства нормализаторами общей стандартизации. При идентификации допускаются вырожденные единичные значения показателей степени. Уровень срабатывания равен 0.

6. Устранение логарифма в показателе степени.

   $\forall_{abcdefg}(0 < a \;\&\; \neg(a - 1 = 0) \;\&\; 0 < b \to a^{g+f(c\log_a b+d)/e} = b^{fc/e}a^{g+fd/e})$

   Выражение f не содержит подвыражения $\log_a b$. Допускаются вырожденные единичные значения c, f, а также нулевое значение g. Антецеденты обрабатываются проверочными операторами. Уровень срабатывания равен 0.

   $\forall_{abcde}(0 < a \;\&\; \neg(a - 1 = 0) \;\&\; 0 < b \to a^{c+d\log_a b/e} = b^{d/e}a^c)$

   Допускаются вырожденные единичные значения d, e и нулевое значение c. Уровень срабатывания равен 0.

   $\forall_{abcde}(0 < a \;\&\; \neg(\log_b a = 0) \;\&\; 0 < b \;\&\; \neg(b - 1 = 0) \to a^{c+d/e\log_b a} = b^{d/e}a^c)$

   Допускаются вырожденные значения c, e. Уровень срабатывания равен 1.

   $\forall_{abcde}(0 < a \;\&\; 0 < b \;\&\; 1 < c \to a^{d(\log_a b)^c/e} = b^{d(\log_a b)^{c-1}/e})$

   Переменная c идентифицируется с натуральной константой. Первые два антецедента обрабатываются проверочными операторами, третий - выделен указателем "программа". Либо преобразуемая степень является сомножителем произведения, имеющего также сомножитель вида b^A, либо редактируется ответ задачи на описание. Допускаются вырожденные значения d, e. Уровень срабатывания равен 2.

$\forall_{abcdefg}(0 < a \ \& \ \neg(a-1=0) \ \& \ 1 < b \ \& \ \log_a b - c\log_a d - e = 0 \to b^{f/g(c\log_a d+e)} = a^{f/g})$

Переменные b, c, d идентифицируются с натуральными константами, переменная e - с неотрицательной целочисленной константой. Либо e есть ноль, либо a - константное выражение. Первые два антецедента обрабатываются проверочными операторами, третий - выделен указателем "программа". Последний антецедент выделен указателем "идентификатор", его левая часть обрабатывается нормализаторами общей стандартизации. Допускаются вырожденные значения c, g, e. Уровень срабатывания равен 2.

$\forall_{abcd}((b/a)^{c\log_{a/b} d} = 1/d^c)$

$\forall_{abcde}(0 < a \ \& \ 0 < b \ \& \ 0 < c \ \& \ \neg(ab-1=0) \to a^{d\log_{ab} c+e} b^{d\log_{ab} c+e} = a^e b^e c^d)$

$\forall_{abcdef}(\neg(b-c=0) \to (b/c)^{e+a/f\log_d(c/b)} = (b/c)^e/d^{a/f})$

$\forall_{abcdef}(\neg(c-e=0) \to (e/c)^{f+a\log_{c/e} d/b} = (e/c)^f/d^{a/b})$

Антецеденты обрабатываются проверочными операторами. Уровень срабатывания равен 1.

$\forall_{abcde}(|a/b|^{c\log_{|b/a|} e/d} = 1/e^{c/d})$

$\forall_a(\exp\ln a = a)$

Уровень срабатывания равен 0.

7. Степенное выражение в основании логарифма.

   $\forall_{abc}(0 < a \ \& \ \neg(b=0) \ \& \ \neg(a^b-1=0) \to \log_{a^b} c = \log_a c/b)$

   Антецеденты обрабатываются проверочными операторами. Указатель "вывод(не(равно(плюс(x1 минус(1))0))эквивалентно(3))" добавляет в контекст замены утверждение $\neg(a-1=0)$, необходимое для сопровождения по о.д.з. Если при этом утверждение $\neg(a^b-1=0)$ далее для сопровождения по о.д.з. не нужно, то оно удаляется. Реализовано это так: сначала оператор "учетвывода" регистрирует информацию о дополнительном утверждении в комментарии (коррекцияпосылок ...), а затем оператор "заменавхождения" использует эту информацию для изменения контекста. Уровень срабатывания равен 0.

   $\forall_{abc}(b - \mathrm{rational} \ \& \ \text{числитель}(b) - \mathrm{even} \ \& \ \neg(a^b-1=0) \to \log_{a^b} c = \log_{|a|} c/b)$

   Антецеденты обрабатываются проверочными операторами. Логарифм входит в условие задачи, причем это условие уже содержит выражение $|a|$. Для сопровождения по о.д.з. вводится дополнительное утверждение $\neg(|a|-1=0)$. Уровень срабатывания равен 1.

   $\forall_{bde}(\neg(d^e-1=0) \ \& \ d < 0 \ \& \ \text{числитель}(e) - \mathrm{even} \ \& \ e - \mathrm{rational} \to \log_{d^e} b = \log_{-d} b/e)$

   Для сопровождения по о.д.з. вводятся утверждения $\neg(e=0), \neg(d+1=0)$. Уровень срабатывания равен 2.

   $\forall_{bde}(\neg(d^e+1=0) \ \& \ \neg(\text{знаменатель}(e) - \mathrm{even}) \ \& \ \neg(\text{числитель}(e) - \mathrm{even}) \ \& \ d < 0 \ \& \ e - \mathrm{rational} \to \log_{-d^e} b = \log_{-d} b/e)$

   Для сопровождения по о.д.з. вводится утверждение $\neg(d+1=0)$. Уровень срабатывания равен 1.

8. Перестановка основания логарифма и выражения под логарифмом.

   $\forall_{ab}(0 < a \ \& \ \neg(a-1=0) \ \& \ 0 < b \ \& \ \neg(b-1=0) \rightarrow \log_b a = 1/\log_a b)$

   Антецеденты обрабатываются проверочными операторами. Заменяются все вхождения логарифма в задачу. Для выполнения замены необходимо выполнение следующих дополнительных требований:

   (a) Неверно, что выражение a содержит неизвестные, а выражение b - не содержит.
   (b) Не имеет места этап редактирования ответа задачи на преобразование.
   (c) Текущий терм задачи имеет вхождение логарифма по основанию a.
   (d) Выражение a не имеет заголовков "дробь", "умножение", "степень".
   (e) Либо выражение b имеет один из заголовков "дробь", "умножение", "степень", либо a не содержит неизвестных, а b - содержит, либо выражение a предшествует в лексикографическом порядке выражению b.

   Для сопровождения по о.д.з. вводится утверждение $\neg(\log_a b = 0)$. Если решается задача на описание, причем текущий терм задачи имеет вхождение неизвестной, не расположенное внутри $\log_a b$ либо $\log_b a$, то уровень срабатывания равен 2, иначе он равен 1.

9. Произведение либо дробь в основании логарифма.

   $\forall_{abc}(0 < a \ \& \ \neg(a-1=0) \ \& \ 0 < b \ \& \ \neg(b-1=0) \rightarrow \log_b c = \log_a c/\log_a b)$

   Логарифм входит в условие задачи на преобразование. Выражение b имеет заголовок "дробь" либо "умножение". Указатель "контекст(разряд(корень логарифм x4)равно(x1 первыйтерм(x4)))" идентифицирует a как основание некоторого другого логарифма, входящего в условие задачи. Проверяется, что a не имеет заголовка "дробь" либо "умножение". Проверяется также, что внутри b имеется вхождение выражения a, достижимое переходами только через операции "плюс", "минус", "умножение", "дробь", "модуль", а также через основания степеней. Для сопровождения по о.д.з. вводится утверждение $\neg(\log_a b = 0)$. Уровень срсбатывания равен 2.

   $\forall_{ab}(0 < a \ \& \ \neg(1/a-1=0) \rightarrow \log_{1/a} b = -\log_a b)$

   Для сопровождения по о.д.з. вводится утверждение $\neg(a-1=0)$. Уровень срабатывания равен 1.

10. Приведение подобных членов для логарифмов модулей выражений, отличающихся знаком.

    $\forall_{abcde}(c\log_e|a-b| + d\log_e|b-a| = (c+d)\log_e|a-b|)$

    Уровень срабатывания равен 1.

11. Сокращение общего множителя при сложении логарифма дроби и логарифма произведения. Прием используется в тех случаях, когда не удается усмотреть знаки сомножителей, числителя и знаменателя:

    $\forall_{abcdef}(0 < ab \ \& \ 0 < acd \rightarrow e\log_f(ab) + e\log_f(c/ad) = e\log_f(bc/d))$

    Антецеденты обрабатываются проверочными операторами. Сумма логарифмов расположена в условии задачи. Для сопровождения по о.д.з. вводится утверждение $0 < bcd$. Уровень срабатывания равен 2.

12. Перестановка основания степени с выражением под логарифмом - показателем степени.

    $\forall_{abcd}(0 < a \;\&\; 0 < b \;\&\; 0 < c \;\&\; \neg(b - 1 = 0) \rightarrow a^{d\log_b c} = c^{d\log_b a})$

    Степень расположена в условии задачи на описание, причем выражение c содержит неизвестные, а выражения a, b, d - не содержат. Антецеденты обрабатываются проверочными операторами. Уровень срабатывания равен 4.

    $\forall_{abcde}(0 < b \rightarrow a^c b^{d\log_e a} = a^{c+d\log_e b})$

    Если выражение расположено в условии задачи на описание, причем b содержит неизвестные, а a, c - не содержат, то прием блокируется. Уровень срабатывания равен 0.

    $\forall_{abcd}(0 < a \rightarrow a^{d\log_b c} = c^{d\log_b a})$

    Степень входит в условие задачи на преобразование, уже имеющее вхождение выражения $\log_b a$. Выражение a не содержит неизвестных. Прием вводит комментарий (перегруппировка $\log_b a$), причем перед выполнением замены проверяется отсутствие комментария (перегруппировка $\log_b c$). Таким образом предотвращается зацикливание. Уровень срабатывания равен 4.

    $\forall_{abcdkmnpqr}(0 < a \;\&\; 0 < b \;\&\; \neg(a - 1 = 0) \;\&\; \neg(b - 1 = 0) \;\&\; r - k = 1 \rightarrow pa^{c+m(\log_b a)^k/n}/qb^{d+m(\log_b a)^r/n} = pa^c/qb^d)$

    Первые четыре антецедента обрабатываются проверочными операторами, пятый - выделен указателем "идентификатор". Выражения k, m, n, p, q могут вырождаться в единицу, выражения c, d - в ноль. Допустима перестановка числителя и знаменателя. Уровень срабатывания равен 3.

    $\forall_{abcdmnpqkr}(\neg(a - 1 = 0) \;\&\; r + k = 1 \rightarrow pa^{c+m/n(\log_b a)^k}/qb^{d+m(\log_b a)^r/n} = pa^c/qb^d)$

    Первый антецедент обрабатывается проверочным оператором, второй - выделен указателем "идентификатор". В остальном - аналогично предыдущему приему.

13. Произведение логарифмов, основание первого из которых равно выражению под вторым.

    $\forall_{abc}(\log_a b \log_b c = \log_a c)$

    Уровень срабатывания равен 0.

14. Попытка разложения на множители суммы под логарифмом.

    $\forall_{abcd}(d = b + c \;\&\; 0 < b + c \rightarrow \log_a(b + c) = \log_a d)$

    Логарифм расположен в условии задачи на описание либо на преобразование, или в посылке задачи на исследование. Сумма под логарифмом не должна иметь дробных слагаемых (сложение дробей под логарифмом выполняется другим приемом). В случае задачи на описание либо на исследование эта сумма должна содержать неизвестные. Первый антецедент выделен указателем "идентификатор". Он предпринимает попытку разложения на множители суммы под логарифмом с помощью нормализатора "видумножение". Проверяется, что результат d не имеет заголовков "плюс", "дробь". Второй антецедент обрабатывается проверочным оператором. Чтобы проверка предпринималась до разложения на множители, используется указатель "копия(фикс(2 2))". Если логарифм находится под описателем "отображение", то прием блокируется. Для сопровождения по о.д.з. выводится утверждение $0 < d$. Уровень срабатывания приема равен 1.

$\forall_{abcd}(d = b + c \ \& \ \neg(b + c = 0) \to \log_a |b + c| = \log_a |d|)$

Аналогично предыдущему, но требования на d ослаблены: оно лишь не должно получаться из исходной суммы перестановками операндов коммутативных операций. Для сопровождения по о.д.з. выводится утверждение $\neg(d = 0)$.

15. Сложение дробей под логарифмом.

    $\forall_{abcde}(d = b/e + c \ \& \ 0 < b/e + c \to \log_a(b/e + c) = \log_a d)$

    Аналогично предыдущему пункту, но требование отсутствия дробных слагаемых отсутствует. Выражение d не должно получаться из исходной суммы перестановкой операндов коммутативных операций. Для сопровождения по о.д.з. выводится посылка $0 < d$. Уровень срабатывания равен 1.

16. Разложение на множители суммы - основания логарифма.

    $\forall_{abcd}(c = a + b \ \& \ \neg(a + b - 1 = 0) \ \& \ 0 < a + b \to \log_{a+b} d = \log_c d)$

    Первый антецедент выделен указателем "идентификатор". Он обрабатывает основание логарифма нормализатором "видумножение". Проверяется, что результат c имеет один из заголовков "умножение", "дробь", "степень", возможно, со знаком "минус". Для сопровождения по о.д.з. выводятся утверждения $\neg(c - 1 = 0), 0 < c$

17. Сопряженные выражения под логарифмом.

    $\forall_{abcde}(ce + ad = 0 \ \& \ cd + abe = 1 \to \log_{c+a\sqrt{b}}(d + e\sqrt{b}) = -1)$

    Выражения a, b, c, d, e суть десятичные константы. Антецеденты выделены указателем "программа". Уровень срабатывания равен 0.

18. Переход к новому основанию для логарифмов, не содержащих неизвестных. Преобразование выполняется лишь в специальных случаях, когда логарифмы по новому основанию уже имелись в текущем терме задачи:

    $\forall_{abc}(0 < a \ \& \ \neg(a - 1 = 0) \ \& \ 0 < b \ \& \ \neg(b - 1 = 0) \to \log_b c = \log_a c/ \log_a b)$

    Логарифм входит в условие задачи на описание либо на преобразование и не содержит неизвестных. Оба выражения $\log_a b$,$\log_a c$ уже встречаются в этом условии. Не имеет места этап завершающего редактирования ответа. Выражение a не совпадает ни с b, ни с c. В условии отсутствует выражение $\log_b a$. Для сопровождения по о.д.з. выводится утверждение $\neg(\log_a b = 0)$. Уровень срабатывания равен 0.

19. Перенесение иррациональности в знаменатель под логарифм. Преобразование используется, если сопряженное выражение уже встречалось под логарифмом:

    $\forall_{abcde}(d = a^2 b - c^2 \ \& \ \neg(d = 0) \ \& \ 0 < a\sqrt{b} + c \to \log_e(a\sqrt{b} + c) = \log_e(d/(a\sqrt{b} - c)))$

    Логарифм входит в условие задачи на описание, причем выражение под логарифмом содержит неизвестные. Первый антецедент выделен указателем "идентификатор". Он обрабатывает правую часть равенства нормализатором "видумножение". Проверяется, что результат d не содержит неизвестных. Второй и третий антецеденты обрабатываются проверочными операторами. Проверяется, что условие уже содержит логарифм, один из операндов которого равен $a\sqrt{b} - c$ либо $c - a\sqrt{b}$. Чтобы избежать обратного перехода, проверяется также, что результат изменения знака у c лексикографически предшествует c. Для сопровождения по о.д.з. выводится следствие $0 < d(a\sqrt{b} - c)$. Уровень срабатывания равен 2.

20. Суммы логарифмов с модулями.

   $\forall_{abc}(a \ln |b/c| + a \ln |c| = a \ln |b|)$

   Уровень срабатывания равен 1.

   $\forall_{abcd}(d \ln(ab) - d \ln(c|b|) = d \ln |a/c|)$

   $\forall_{abcd}(d \ln |a| + d \ln |c + b/a| = d \ln |ac + b|)$

   Сумма входит в условие задачи на описание, имеющей цель (связка ...). Приемы ориентированы на ситуации, возникающие при решении дифференциальных уравнений. Уровень срабатывания равен 1.

21. Упрощение константных выражений с логарифмами в показателе степени.

   $\forall_{abcdep}(\log_a(dc^e) = p \to a^{b/(\log_c d + e)} = c^{b/p})$

   Преобразуемая степень представляет собой константное выражение. Антецедент выделен указателем "идентификатор". Он обрабатывает левую часть равенства вспомогательной задачей на упрощение. Проверяется, что результат p не содержит символа "логарифм". Уровень срабатывания равен 3.

22. Упрощение константных дробей с логарифмами.

   $\forall_{abcdpqr}((a \log_b c + d)/(p \log_b q + r) = \log_{q^p b^r}(c^a b^d))$

   Выражения a, p, d, r суть целочисленные константы, не превосходящие по модулю 4; выражения b, c, q - десятичные константы. Уровень срабатывания равен 2.

   $\forall_{abcd}(d = a^2 - b \to p \log_{a+\sqrt{b}} c/q(\log_{a+\sqrt{b}} d - 1) = p \log_{a-\sqrt{b}} c/q)$

   Выражения a, b, d суть десятичные константы. Антецедент выделен указателем "программа". Уровень срабатывания равен 3.

23. Логарифмы константных выражений с радикалами.

   $\forall_{abcde}(a^2 + b^2 c = d \;\&\; 2ab = e \to \log_{a+b\sqrt{c}}(d + e\sqrt{c}) = 2)$

   $\forall_{abcdmn}(d = a^2 - b^2 c \;\&\; md^2 - a^2 - b^2 c = 0 \;\&\; nd^2 + 2ab = 0 \to \log_{a+b\sqrt{c}}(m + n\sqrt{c}) = -2)$

   Все переменные идентифицируются с целочисленными константами. Антецеденты выделены указателем "программа". Уровень срабатывания равен 1.

24. Сложение логарифмов с сопряженными радикалами.

   $\forall_{abc}(a \log_b(\sqrt{c+1} - 1) - a \log_b(\sqrt{c+1} + 1) = 2a \log_b(\sqrt{c+1} - 1) - a \log_b c)$

   Уровень срабатывания равен 1.

25. Исключение логарифма с помощью группировки.

   $\forall_{abc}(a \log_{bc} b + a \log_{bc} c = a)$

   Уровень срабатывания равен 2.

$\forall_{abcde}(\neg(c(b-a)-1=0)\ \&\ \text{числитель}(e)-\text{even}\ \&\ e-\text{rational}\rightarrow d\log_{c(b-a)}(c^e)+d\log_{c(b-a)}((a-b)^e)=de)$

Уровень срабатывания равен 1.

26. Сокращение логарифмов с дробными взаимно обратными основаниями.

    $\forall_{abcde}(\neg(c-e=0)\ \&\ \neg(d-1=0)\rightarrow a\log_{c/e}d/b\log_{e/c}d=-a/b)$

    Уровень срабатывания равен 1.

Общая стандартизация утверждений

1. Равенство логарифма нулю.

   $\forall_{ab}(\log_a b=0\leftrightarrow b=1)$

   Равенство не расположено внутри утверждения, используемого для сопровождения по о.д.з. Уровень срабатывания равен 1.

2. Равенство логарифмов.

   $\forall_{abc}(\log_a b-\log_a c=0\leftrightarrow b-c=0)$

   Уровень срабатывания равен 2.

3. Равенство логарифма единице.

   $\forall_{ax}(\log_a x=1\leftrightarrow a=x)$

   Уровень срабатывания равен 0.

Решение уравнений

1. Потенцирование логарифмических уравнений.

   $\forall_{abcde}(0<a\ \&\ \neg(a-1=0)\ \&\ 0<b\rightarrow c\log_a b+d=e\leftrightarrow b^c a^d-a^e=0)$

   Равенство является условием задачи на описание либо посылкой задачи на исследование. Хотя бы одно из выражений a, b содержит неизвестные. Возможные уровни срабатывания приема равны 1,2,3,5. Должны выполняться следующие дополнительные требования:

   (a) Если число неизвестных задачи более одной, то уровень срабатывания не равен 2, иначе - не равен 3.

   (b) Либо отсутствует комментарий "логарифм", блокирующий потенцирование логарифмических уравнений, либо выражения a, c не содержат неизвестных, а каждое содержащее неизвестные слагаемое выражения d имеет вид $k\log_A B$, где k, A известны.

   (c) Либо текущий уровень равен 1 и выражение b содержит степень с неизвестным показателем, либо каждое слагаемое выражения d, пересекающееся по своим неизвестным с $\log_a b$, имеет вид $k\log_a t$, где k известно, причем если текущий уровень равен 1, то решается задача на описание с несколькими неизвестными, не имеющая других уравнений с подвыражением вида $\log_A b$ либо b^A.

   (d) Выражение c не содержит неизвестных логарифмов.

(e) Если решается дифференциальное уравнение, то выражение c не содержит неизвестных и не имеет места этап редактирования ответа.

(f) Если решается задача на отыскание корней производной, причем усматривается целесообразность попытки повторного дифференцирования для доказательства отсутствия корней, то уровень срабатывания равен 5, иначе он не равен 5.

Антецеденты обрабатываются проверочными операторами. Указатель "удалениезамечания(комментарий(новыенеизвестные 2)тип(описать))" исключает комментарий (новыенеизвестные 2), инициируя тем самым повторную попытку перехода к новым неизвестным.

$\forall_{abc}(0 < a \ \& \ \neg(a-1=0) \ \& \ 0 < b \rightarrow \log_a b = c \leftrightarrow b = a^c)$

Равенство является условием задачи на описание либо посылкой задачи на исследование. Выражение b содержит неизвестные, выражения a, c - не содержат. Антецеденты обрабатываются проверочными операторами. Указатель "удалениеусловия(меньше(0 x2))" обеспечивает проверку того, будет ли после выполнения замены использоваться для сопровождения по о.д.з. неравенство $0 < b$. Если нет, то оно удаляется из списка условий. Уровень срабатывания равен 0.

$\forall_{abc}(0 < a \ \& \ \neg(a-1=0) \ \& \ 0 < b \rightarrow \log_a b = c \leftrightarrow \neg(c=0) \ \& \ a = b^{1/c} \ \vee \ c = 0 \ \& \ b = 1)$

Равенство входит в условие задачи на описание либо в посылку задачи на исследование, причем не под отрицанием. Выражение a содержит неизвестные, выражения b, c - не содержат. Уровень срабатывания равен 0.

$\forall_{abc}(0 < a \ \& \ \neg(a-1=0) \rightarrow \log_a b = c \leftrightarrow b = a^c)$

Равенство является условием задачи на редактирование параметрического описания - ответа дифференциального уравнения. Выражение b содержит неизвестные, выражения a, c - не содержат. Уровень срабатывания равен 2.

2. Выбор общего основания неизвестных логарифмов.

При решении уравнений и неравенств с неизвестными логарифмами выполняется переход к общему основанию таких логарифмов. Это делается в два этапа. Сначала выбирается новое основание логарифмов c и создается комментарий (нормлогарифм c) к текущей задаче на описание. Затем срабатывают приемы, использующие этот комментарий для фактического преобразования логарифмов.

Начнем с рассмотрения приемов, выбирающих новое основание. Этих приемов всего два, и оба имеют теорему $\forall_{ab}$(контекст $\log_a b$). Заголовки приемов - "замечание", так как действия их сводятся к созданию комментария (нормлогарифм $\ldots$). Данный вид теоремы означает, что инициализация приема начинается с усмотрения вхождения выражения $\log_a b$. Проверяется, что это вхождение расположено в условии задачи на описание и что отсутствует ранее введенный комментарий (нормлогарифм $\ldots$). Проверяется также выполнение следующих требований:

(a) Логарифм содержит неизвестные.

(b) Либо текущее условие задачи является уравнением, либо задача вообще не имеет уравнений.

   (c) Некоторое условие задачи содержит неизвестный логарифм с основанием, отличным от a.
   (d) Текущее вхождение логарифма не расположено внутри другого логарифма.
   (e) Текущее условие не используется для сопровождения по о.д.з.

   Если эти требования выполнены, то указатель "контекст(обобщмножитель(x1 x3))" далее идентифицирует c с обобщенным множителем выражения a - некоторым подвыражением, достижимым из a при переходах только через операции "минус", "умножение", "дробь", "модуль", а также через основания степеней. Для выбора наиболее удобного нового основания c проверяется еще ряд требований:

   (a) Выражение c не имеет своим заголовком символы "умножение", "дробь", "степень".
   (b) Усматривается положительность выражения c.
   (c) Отсутствует другой неизвестный логарифм, входящий в условия задачи, у которого некоторый сомножитель d основания, положительный и не имеющий своим заголовком символы "умножение", "дробь", "степень", оценивается процедурой "фильтроснования" как более предпочтительный, чем c. Эта процедура реализована на ГЕНОЛОГе в виде оператора фильтра. Она решает, что d предпочтительнее, чем c, если выполнено одно из условий:
      i. Усматривается $d \neq 1$, но не усматривается $c \neq 1$.
      ii. Выражение d известно, а c - содержит неизвестные.
      iii. Выражение d константное, а c - неконстантное.
      iv. Длина выражения d меньше длины c, причем из прочих соображений не вытекает, что c предпочтительнее d.
   (d) Если c содержит неизвестные, а b - не содержит, то в условия задачи должен входить неизвестный логарифм $\log_A B$, у которого выражение B отлично от b. При нарушении данного требования будет применяться второй прием выбора основания (см. ниже).

   Если эти требования выполнены, прием вводит комментарий (нормлогарифм c). Уровень срабатывания равен 2.

   Второй прием ориентирован на остаточный случай - под всеми неизвестными логарифмами находится одно и то же положительное известное выражение b, которое и берется в качестве нового основания. Общие ограничения на контекст срабатывания те же, что и выше. Прием вводит комментарий (нормлогарифм b). Уровень срабатывания равен 2.

   Оба приема уменьшают до 0 веса всех условий задачи, содержащих неизвестные логарифмы. Это позволяет сразу сработать приему, выполняющему переход к новому основанию этих логарифмов.

3. Преобразование логарифмов к новому основанию.

   Если уже создан комментарий (нормлогарифм a), то неизвестные логарифмы преобразуются к основанию a:

$\forall_{abc}(0 < a \mathbin{\&} \neg(a - 1 = 0) \mathbin{\&} 0 < b \mathbin{\&} \neg(b - 1 = 0) \to \log_b c = \log_a c / \log_a b)$

Логарифм содержит неизвестные и входит в условие задачи на описание либо в посылку задачи на исследование, не используемые для сопровождения по о.д.з. Указатель "контекст(комментарий(нормлогарифм x1))" идентифицирует новое основание a. Проверяется, что оно отлично от b и что существует вхождение неизвестной в текущее условие задачи, не расположенное внутри выражения $\log_b c$. Проверяется также, что преобразуемый логарифм не расположен внутри другого логарифма. Антецеденты обрабатываются проверочными операторами. Для сопровождения по о.д.з. выводится утверждение $\neg(\log_a b = 0)$. Уровень срабатывания равен 0.

4. Разбор случаев для сравнения нового основания логарифмов с единицей.

   Вообще говоря, не гарантировано, что новое основание логарифма окажется не равным единице. Если это не усматривается, то предпринимается разбор случаев:

   $\forall_a(a = 1 \ \vee \ \neg(a - 1 = 0))$

   Прием имеет заголовок "выводусловия". Указатель "контрольвывода(логарифм(x2 x3))" инициирует его применение при усмотрении какого-либо логарифма в условии задачи на описание. Указатель "контекст(комментарий(нормлогарифм x1))" идентифицирует новое основание a. Проверяется, что утверждение $\neg(a - 1 = 0)$ не усматривается проверочным оператором. Выводимая дизъюнкция снабжается комментарием "разборслучаев". Уровень срабатывания равен 0.

5. Выявление повторного вхождения логарифма неизвестной дроби.

   Если под логарифмом находится дробное выражение, знаки числителя и знаменателя которого установить не удается, то оно не преобразуется в разность логарифмов. Тогда могут оказаться полезными приемы, преобразующие другие логарифмы так, чтобы возникали повторные вхождения данного логарифма дроби. Это подготавливает возможность перехода к новым неизвестным.

   $\forall_{abcd}(0 < a/b \to \log_c(a^d/b^d) = d \log_c(a/b))$

   $\forall_{abcde}(0 < ab \to e \log_c(a^d) - e \log_c(b^d) = ed \log_c(a/b))$

   Логарифм входит в уравнение, являющееся условием задачи на описание. Хотя бы одно из выражений a, b содержит неизвестные. Задача имеет уравнение, содержащее логарифм дроби a/b либо b/a. Во втором приеме допускаются вырожденные единичные значения d, e. Антецедент обрабатывается проверочным оператором. Уровень срабатывания равен 0.

6. Группировка неизвестных логарифмов.

   $\forall_{abcde}(a \ln b + c \ln b + d = e \leftrightarrow (a + c) \ln b + d = e)$

   Уравнение входит в условие задачи на описание, решаемой для нахождения корней производной. Выражение b содержит неизвестные. Каждое вхождение неизвестной в выражения a, c, d достижимо из корня этого выражения переходами только через операции "плюс", "минус", "умножение", "дробь", "модуль", а также через основания степеней. Уровень срабатывания равен 2.

   $\forall_{abcde}(a \ln |b/c| + d = -a \ln(ec) + f \leftrightarrow a \ln |be| + d = f)$

Уравнение является условием задачи на описание, имеющей цель (связка ...). Уровень срабатывания равен 3.

7. Вывод следствий.

   $\forall_{abcde}(a \log_b c + d = e \to c^a b^d = b^e)$

   Уравнение является посылкой задачи на исследование, имеющей цель "известно". Выражение d содержит неизвестные. Существует посылка задачи с подвыражением вида b^p, где p содержит неизвестные. Для блокировки повтора используется комментарий (логарифм степень) к текущей посылке. Уровень срабатывания равен 4.

Активизация попытки перехода к новым неизвестным

Попытка перейти к новым неизвестным предпринимается в начале решения уравнений либо неравенств. Если она была неудачна, то в процессе преобразований целесообразно время от времени предпринимать ее снова. Это делается путем исключения комментария (новыенеизвестные ...), блокирующего повторение попытки. Заметим, что многие из приведенных выше приемов, преобразующих уравнения с логарифмами, такой комментарий удаляли. Приведем прием, который специально предназначен для этого. Он срабатывает, если оказывается, что все вхождения неизвестных заключены внутри вхождений одного и того же логарифма. Теорема приема имеет вид $\forall_{ab}$(контекст$(\log_a b)$). Заголовок приема - "замечание". Прием инициируется при усмотрении вхождения неизвестного логарифма $\log_a b$ в неравенство, являющееся условием задачи на описание. Проверяется, что задача не имеет уравнений и что все вхождения неизвестных в текущее неравенство расположены только внутри вхождений выражения $\log_a b$. Проверяется также, что число последних вхождений более одного и что текущее вхождение логарифма находится в знаменателе некоторой дроби. Для блокировки повтора используется комментарий (логарифм новыенеизвестные $\log_a b$) к текущей посылке. Уровень срабатывания равен 2.

Нормализатор общей стандартизации "нормлогарифм"

В нормализаторе собраны копии следующих приемов общей стандартизации:

1. Логарифм единицы.
2. Число под логарифмом равно основанию логарифма.
3. Степенное выражение в основании логарифма.
4. Логарифм произведения.
5. Логарифм дроби.

   $\forall_{abc}(0 \le a \;\&\; 0 \le b \to \log_c(a/b) = \log_c a - \log_c b)$

   $\forall_{abc}(a \le 0 \;\&\; b \le 0 \to \log_c(a/b) = \log_c(-a) - \log_c(-b))$

   $\forall_{ab}(0 < a\_\& \; \neg(a/b - 1 = 0) \to \log_{a/b}(b/a) = -1)$

   $\forall_{ab}(\log_a(1/b) = -\log_a b)$

6. Логарифм степени.

   $\forall_{abc}(0 \le a \to \log_b(a^c) = c \log_a b)$

7. Логарифм модуля степени.

   $\forall_{abc}(c - \text{rational} \ \& \ \neg(\text{знаменатель}(c) - \text{even}) \rightarrow \log_b(|a^c|) = c \log_b |a|)$

8. Дробь с числителем единица в основании логарифма.

   $\forall_{ab}(0 < a \ \& \ \neg(1/a - 1 = 0) \rightarrow \log_{1/a} b = -\log_a b)$

9. Приведение всех встречающихся в преобразуемом терме логарифмов к основанию, указанному в комментарии (новыенеизвестные ...). Этот комментарий нормализатору не передается, а извлекается из текущей задачи.

10. Символы бесконечности.

    $\ln 0 = -\infty$

    $\ln \infty = \infty$

    Предполагается, что указанные выражения могут передаваться нормализатору лишь в тех случаях, когда их использование, в строго ограниченном контексте, является корректным.

Нормализатор стандартизации логарифмов с неизвестными "уравнлогарифм"

Прежде всего, нормализатор дублирует следующие приемы общей стандартизации:

1. Логарифм единицы.
2. Число под логарифмом равно основанию логарифма.
3. Степенное выражение в основании логарифма.
4. Дробь с числителем единица в основании логарифма.
5. Логарифм произведения (сомножители неотрицательны).
6. Логарифм дроби (числитель и знаменатель неотрицательны).
7. Логарифм степени.

Кроме этого, добавлены следующие приемы:

1. Обращение к стандартизации основания логарифма.

   $\forall_{abc}(c = a \ \& \ 0 < a \ \& \ \neg(a - 1 = 0) \rightarrow \log_a b = \log_c b)$

   Прием применяется при наличии комментария "нормуравн". Выражение a неконстантное. Первый антецедент, выделенный указателем "идентификатор", обращается к процедуре "нормуравн". Проверяется, что результат c отличен от a. Второй и третий антецеденты обрабатываются проверочными операторами. Для сопровождения по о.д.з. выводятся утверждения $0 < c, \neg(c - 1 = 0)$. Нормализатор регистрирует их в комментарии (коррекцияпосылок ...), откуда они впоследствии будут перенесены в текущую задачу.

2. Обращение к стандартизации выражения под логарифмом.

   $\forall_{abc}(c = b \ \& \ 0 < b \rightarrow \log_a b = \log_a c)$

   Аналогично предыдущему.

3. Перестановка основания логарифма и выражения под логарифмом.

   $\forall_{ab}(0 < a \ \& \ \neg(a - 1 = 0) \ \& \ 0 < b \ \& \ \neg(b - 1 = 0) \rightarrow \log_b a = 1/\log_a b)$

   Исходный обрабатываемый оператором "нормуравн" терм уже содержит выражение $\log_a b$, но не содержит неизвестных логарифмов, отличных от $\log_a b, \log_b a$. Выражение a не имеет своим заголовком символы "дробь", "умножение", "степень". Выражение b содержит неизвестные. Либо выражение a не содержит неизвестных, либо выражение b имеет своим заголовком один из символов "умножение", "дробь", "степень", либо выражение a предшествует выражению b в лексикографическом порядке. Для сопровождения по о.д.з. выводится утверждение $\neg(\log_a b = 0)$.

4. Сложение дробей под логарифмом.

   $\forall_{abcde}(d = b/c + e \ \& \ 0 < b/e + c \rightarrow \log_a(b/e + c) = \log_a d)$

   Основание логарифма a не содержит неизвестных, а выражение под логарифмом - содержит. Первый антецедент выделен указателем "идентификатор" и обрабатывает правую часть равенства нормализатором "видумножение". Второй антецедент обрабатывается проверочным оператором. Для сопровождения по о.д.з. выводится утверждение $0 < d$.

5. Перенесение иррациональности в знаменатель для выражения под логарифмом.

   $\forall_{abcde}(d = a^2b - c^2 \ \& \ \neg(d = 0) \ \& \ 0 < a\sqrt{b} + c \rightarrow \log_e(a\sqrt{b} + c) = \log_{(} d/(a\sqrt{b} - c)))$

   Исходный обрабатываемый оператором "нормуравн" терм имеет логарифм, под которым находится $a\sqrt{b} - c$ или $c - a\sqrt{b}$, либо основание которого равно одному из этих выражений. Выражение $a\sqrt{b} + c$ содержит неизвестные. Первый антецедент выделен указателем "идентификатор" и обрабатывает правую часть равенства нормализатором раскрывания скобок "стандплюс". Результат d не содержит неизвестных. Выражение $-c$ (после обработки нормализатором "нормминус") лексикографически предшествует выражению c. Второй и третий антецеденты обрабатываются проверочными операторами. Для сопровождения по о.д.з. выводится утверждение $0 < d(a\sqrt{b} - c)$.

Глава 11

Приемы, связанные с тригонометрическими функциями

Значительная часть приемов, связанных с тригонометрическими функциями, уже встречалась в рассмотренных выше пакетных операторах "видумножение" и "стандплюс". Здесь мы приводим остальные приемы.

11.1 Приемы символа "синус"

Выражение "синус(a)", прорисовываемое формульным редактором в виде $\sin a$, обозначает синус вещественного числа a. Заметим, что общепринятые обозначения $\sin^m a$, $\cos^m a$ и т.п. для степеней тригонометрических функций формульным редактором не поддерживаются. Вместо этого применяются обозначения $(\sin a)^m$, $(\cos a)^m$, и т.д. Справочники "тип", "арность", "одз", "типданных" характеризуют простейшие свойства символа "синус".

Общая стандартизация выражений

1. Синус константы.

   (a) Синус нуля.
   $\sin 0 = 0$

   (b) Синус половины пи.
   $\sin \pi/2 = 1$

   (c) Синус одной шестой пи.
   $\sin \pi/6 = 1/2$

   (d) Синус одной четвертой пи.
   $\sin \pi/4 = 1/\sqrt{2}$

   (e) Синус одной третьей пи.
   $\sin \pi/3 = \sqrt{3}/2$

   Уровень срабатывания приведенных приемов равен 0.

   (f) Синус одной пятой пи.
   $\sin \pi/5 = \sqrt{10 - 2\sqrt{5}}/4$

Если решается задача на преобразование, имеющая цель "известны" (т.е. происходит редактирование ответа задачи на описание, имеющей цель "известно ..."), то уровень срабатывания равен 3, иначе он равен 6. Выполняется одновременная замена всех вхождений в задачу данного синуса. Цель "элемсвертка" задачи на описание блокирует прием.

(g) Синус одной десятой пи.
$\sin \pi/10 = (\sqrt{5} - 1)/4$

(h) Синус одной восьмой пи.
$\sin \pi/8 = \sqrt{2 - \sqrt{2}}/2$

(i) Синус одной двенадцатой пи.
$\sin \pi/12 = (\sqrt{3} - 1)/2\sqrt{2}$

Уровни срабатывания перечисленных приемов - аналогично случаю синуса одной пятой пи. Кроме того, создана копия приема для синуса одной двенадцатой пи, применяемая при редактировании ответа задачи на описание, если преобразуемый терм содержит $\sqrt{3}$ и имеет большую длину. Эта копия срабатывает на уровне 1.

(j) Произведение константных синусов.
$\sin 2\pi/9 \sin \pi/18 = (\sqrt{3} - 2\sin 2\pi/9)/4$

Замена выполняется в условии задачи на преобразование, имеющей цель "известны". Уровень срабатывания равен 3.

$\forall_a(\sin(\pi/4 + a)\cos(\pi/4 + a) = \cos(2a)/2)$

Выражение a константное. Уровень срабатывания равен 0.

2. Синус целого кратного половины пи.

   $\forall_n(n - \text{целое} \rightarrow \sin(\pi n/2) = (0 \text{ при } n - \text{even}, \text{ иначе } (-1)^{(n-1)/2}))$

   Антецедент обрабатывается проверочным оператором. Уровень срабатывания равен 4.

3. Минус под синусом.

   $\forall_a(\sin(-a) = -\sin a)$

   Уровень срабатывания равен 0.

4. Аргумент домножен на степень минус единицы.

   $\forall_{abn}(n - \text{целое} \rightarrow \sin(a(-1)^n/b) = (-1)^n \sin(a/b))$

   Уровень срабатывания равен 0.

5. Изменение знака разности под синусом.

   $\forall_{abc}(c = a - b \rightarrow \sin(b - a) = -\sin c)$

   Антецедент выделен указателем "идентификатор" и обрабатывает правую часть равенства нормализаторами "нормплюс", "стандупорядочение". Проверяется, что результат c лексикографически предшествует выражению $b-a$. Прием блокируется, если сумма под синусом имеет слагаемое вида $k\pi/n$, где k - целочисленная, а n - натуральная константа. Он также блокируется при завершающем редактировании ответа задачи на преобразование. Уровень срабатывания равен 1. По той же теореме создана еще одна версия приема, используемая для завершающего редактирования ответа задачи на преобразование. Здесь требуется наличие цепочки переходов от преобразуемого синуса к внешнему символу "минус", идущей только через символы "умножение", "дробь", "модуль"и через основания степеней. Уровень срабатывания равен 2.

$\forall_{abc}(-a\sin(b-c) = a\sin(c-b))$

Указатель "эквивалентно" означает, что прием используется не для тождественного преобразования, а для замены равенства на константу "истина". Он применяется в задачах на исследование. Уровень срабатывания равен 0.

6. Преобразование дробного слагаемого аргумента к виду суммы дробей. В решателе принята стандартизация аргументов тригонометрических операций путем преобразования их к виду явных сумм:

   $\forall_{abcdefn}(f = (a+b)^n c/d + e \rightarrow \sin((a+b)^n c/d + e) = \sin f)$

   Показатель степени n идентифицируется с натуральной константой, не превосходящей 3, причем допускается вырожденное единичное значение. В единицу могут также обращаться параметры c, d, однако хотя бы один из параметров n, c, d должен быть отличен от единицы. Допускается вырожденное нулевое значение параметра e. Антецедент выделен указателем "идентификатор" и преобразует правую часть равенства нормализатором раскрывания скобок "стандплюс". Прием блокируется при решении задачи на описание, если выражения a, b не содержат неизвестных, а c - содержит. Он блокируется также в задачах на преобразование, имеющих цель "учетрезультата" (т.е. при завершающем редактировании ответа задач на описание, имеющих цель (известно ...)). После применения приема в задаче на описание активируется повторная попытка перехода к новым неизвестным. Уровень срабатывания равен 1.

7. Формулы приведения.

   $\forall_{abcde}(b = cd + e \;\&\; 0 < c \rightarrow \sin(a + b\pi/d) = (-1)^c \sin(a + e\pi/d))$

   Здесь и в последующих приемах коэффиценты в дроби с π идентифицируются с натуральными константами. Антецеденты выделены указателем "программа". Прием делает коэффициент при π меньшим единицы. Уровень срабатывания равен 0.

   $\forall_{abcd}(b = 2a - c \;\&\; 0 \leq b \rightarrow \sin(d + a\pi/c) = \cos(d + b\pi/2c))$

   Прием делает коэффициент при π меньшим одной второй. Уровень срабатывания равен 0.

   $\forall_{abcdefg}(b = cd + e \;\&\; f = c - e \;\&\; g = d + 1 \rightarrow \sin(a - b\pi/c) = (-1)^g \sin(a + f\pi/c))$

   Прием переходит от отрицательного коэффициента при π к положительному, не превосходящему единицы. Уровень срабатывания равен 0.

   $\forall_{abcde}(c = 4e - b \;\&\; 0 \leq c \;\&\; d = b - 2e \rightarrow \sin(a + e\pi/b) = \cos(-a + d\pi/2b))$

   Прием делает коэффициент при π не превосходящим одной четвертой. Для коэффициента, равного одной четвертой, замена выполняется, если либо a имеет заголовок "минус", либо обработанный нормализатором терм $-a$ лексикографически предшествует терму a. Уровень срабатывания равен 1.

   Заметим, что описанный выше прием изменения знака разности под синусом будет в рассматриваемом случае блокироваться, так как имеется слагаемое с рациональным кратным π. Поэтому коэффициент при π останется положительным, и зацикливание не возникнет.

$\forall_{an}(n - \text{целое} \rightarrow \sin(a + n\pi) = (-1)^n \sin a)$

Здесь не требуется, чтобы n было целочисленной константой. Антецедент обрабатывается проверочным оператором. Уровень срабатывания равен 0.

8. Стандартизация тригонометрических сумм с применением тождеств для суммы квадратов либо четвертых степеней синуса и косинуса. Эти приемы уже встречались в нормализаторе "видумножение", где были приведены пояснения к их теоремам.

   $\forall_{abcdefghijxl}(a - b = 2 \;\&\; c - d = 2 \;\&\; (0 < e \;\&\; 0 < f \;\&\; g = \min(e, f) \vee e < 0 \;\&\; f < 0 \;\&\; g = \max(e, f)) \;\&\; h = e - g \;\&\; i = f - g \rightarrow ej(\cos x)^c(\sin x)^b + fj(\cos x)^d(\sin x)^a + l = hj(\cos x)^c(\sin x)^b + ij(\cos x)^d(\sin x)^a + gj(\cos x)^d(\sin x)^b + l)$

   Переменные a, b, c, d идентифицируются с натуральными константами, переменные e, f - с десятичными константами. Допускаются вырожденные нулевые значения b, d. Уровень срабатывания равен 1.

   $\forall_{xbcdefghij}(f - e = 2 \;\&\; (0 < c \;\&\; d < 0 \;\&\; g = \min(c, -d) \vee c < 0 \;\&\; 0 < d \;\&\; g = \max(c, -d)) \;\&\; h = c - g \;\&\; i = d + g \rightarrow cb(\cos x)^e + db(\cos x)^f + j = hb(\cos x)^e + ib(\cos x)^f + gb(\sin x)^2(\cos x)^e + j)$

   Переменная e идентифицируется с целочисленной константой, f - с натуральной. Выражения c, d - десятичные константы. Выражение a неконстантное. Допускаются вырожденные единичные значения b, c, d, e, f. Либо h, i равны 0, либо текущий терм задачи должен содержать тригонометрическюу операцию, отличную от $\cos a$. Если преобразуемая сумма входит в уравнение либо неравенство с нулевой противоположной частью и является "обобщенным множителем" своей части (т.е. достижима из ее корня переходами только через символы "минус", "умножение", "дробь", "модуль" и через основания степени), то прием блокируется. Уровень срабатывания равен 1.

   $\forall_{xbcdefghij}(d - c = 2 \;\&\; (0 < e \;\&\; f < 0 \;\&\; g = \min(e, -f) \vee e < 0 \;\&\; 0 < f \;\&\; g = \max(e, -f)) \;\&\; h = e - g \;\&\; i = f + g \rightarrow eb(\cos x)^c + fb(\cos x)^d + j = hb(\sin x)^c + ib(\sin x)^d + gb(\cos x)^2(\sin x)^c + j)$

   Аналогично предыдущему.

   $\forall_{abcdef}(b = c \;\&\; d - c = 4 \rightarrow a(\sin e)^d(\cos e)^c + a(\sin e)^b(\cos e)^d + f = a(\sin e)^c(\cos e)^c - 2a(\sin e)^{c+2}(\cos e)^{c+2} + f)$

   Переменные b, c, d идентифицируются с целочисленными константами. Уровень срабатывания равен 1.

   $\forall_{abx}(a(\cos x)^2/b(\sin x)^2 - a/b(\sin x)^2 = -a/b)$

   Уровень срабатывания равен 1.

9. Выражение частного одинаковых степеней синуса и косинуса через тангенс и котангенс.

   $\forall_{abcd}(c - \text{rational} \;\&\; \neg(\text{знаменатель}(c) - \text{even}) \rightarrow a(\sin b)^c/d(\cos b)^c = a(\operatorname{tg} b)^c/d)$

   $\forall_{abcd}(c - \text{rational} \;\&\; \neg(\text{знаменатель}(c) - \text{even}) \rightarrow a(\cos b)^c/d(\sin b)^c = a(\operatorname{ctg} b)^c/d)$

Дробь является условием задачи на преобразование, имеющей цель "разложитьнамножители". Уровень срабатывания равен 1.

$\forall_{abcdepq}(p = \min(b, c) \to d(\cos a)^b(\operatorname{ctg} a)^q/(sina)^c e = (\operatorname{ctg} a)^{p+q}(\cos a)^{b-p}d/(\sin a)^{c-p}e)$

$\forall_{abcdepq}(p = \min(b, c) \to d(\sin a)^b(\operatorname{tg} a)^q/(cosa)^c e = (\operatorname{tg} a)^{p+q}(\sin a)^{b-p}d/(\cos a)^{c-p}e)$

Дробь входит в условие задачи на преобразование, имеющей цель "упростить". Переменные b, c идентифицируются с натуральными константами. Выражения b, c, d, e, q могут вырождаться в единицу. Уровень срабатывания равен 4.

$\forall_a(\sin a/\cos a = \operatorname{tg} a)$

$\forall_a(\cos a/\sin a = \operatorname{ctg} a)$

Дробь расположена внутри арктангенса либо арккотангенса. Уровень срабатывания равен 1.

$\forall_a(\sin a/\cos a = \operatorname{tg} a)$

Указатель "эквивалентно" определяет преобразование замены данного равенства на константу "истина". Уровень срабатывания равен 0.

10. Сумма шестых степеней синуса и косинуса.

    $\forall_{ab}(b(\sin a)^6 + b(\cos a)^6 = b - 3b(\sin a)^2(\cos a)^2)$

    Уровень срабатывания равен 0.

11. Усмотрение сокращений при переходе к косинусу либо синусу половинного аргумента.

    $\forall_{abc}(c = 2b \to a\cos c + 2a(\sin b)^2 = a)$

    Антецедент выделен указателем "идентификатор". Чтобы получить представление тригонометрического аргумента в стандартном виде, он обрабатывает правую часть равенства нормализатором "стандплюс". Уровень срабатывания равен 2.

    $\forall_{abpq}(b - 2a = 0 \ \&\ 0 \leq b \ \&\ 0 \leq \pi - b \to p\sin a/q\sqrt{1 - \cos b} = p/q\sqrt{2})$

    Прием применяется в посылках задач на доказательство либо на исследование. Первый антецедент выделен указателем "идентификатор", остальные - обрабатываются проверочными операторами, и для них введены сильные ограничители трудоемкости. Уровень срабатывания равен 0.

12. Модуль под синусом.

    $\forall_a(\sin|a| = \sin a \mathrm{sg} a)$

    Прием применяется в посылках задач на исследование, имеющих цель "известно". Выражение a должно содержать неизвестные. Уровень срабатывания равен 2.

13. Сумма произведения тригонометрических функций и функции от суммы либо разности аргументов.

    $\forall_{abc}(c\sin b\cos a + c\sin(a - b) = c\sin a\cos b)$

    $\forall_{abc}(c\sin(a - b) - \ \sin a\cos b = -c\sin b\cos a)$

    $\forall_{abc}(b\sin(a + c) - b\sin c\cos a = b\sin a\cos c)$

    Уровень срабатывания равен 0.

14. Приведение подобных членов с переходом к синусу двойного угла.

    $\forall_{apqn}(p(\sin a)^n(\cos a)^n + q(\sin(2a))^n = (p + 2^n q)(\sin(2a))^n/2^n)$

    Коэффициенты p, q не должны содержать тригонометрических операций. Допускаются вырожденные единичные значения p, q, n. Уровень срабатывания равен 3.

15. Преобразование подынтегрального выражения. Приведем несколько приемов, используемых при формальном интегрировании для предварительного преобразования подынтегрального выражения.

    (a) Понижение степени числителя путем выражения квадрата синуса через квадрат косинуса.

    $\forall_{abcde}(a(\sin e)^c/b(\cos e)^d = a(\sin e)^{c-2}/b(\cos e)^d - a(\sin e)^{c-2}/b(\cos e)^{d-2})$

    Дробь входит в условие задачи на преобразование, имеющей цель "нормИнтеграл". Она расположена только внутри надвыражений с заголовками "плюс", "минус", "умножение". Переменная c идентифицируется с натуральной константой, большей единицы. Указатель "контекст(комментарий(нормИнтеграл x6))" извлекает из комментариев задачи переменную интегрирования x6. Проверяется, что она входит в выражение e. Уровень срабатывания равен 1.

    (b) Преобразование к виду суммы тригонометрических выражений.

    $\forall_{abcdep}(p = a(\sin b)^c/d + e \to a(\sin b)^c/d + e = p)$

    Сумма является условием задачи на преобразование, имеющей цель "нормИнтеграл". Антецедент выделен указателем "идентификатор" и обрабатывает правую часть равенства нормализатором "стандплюс", снабженным комментарием "видумножение". Такой комментарий активирует приемы перехода от тригонометрических произведений к суммам. Допускаются вырожденное нулевое значение переменной e и единичные значения a, d. Переменная интегрирования должна входить в b и не входить в d. Она не должна встречаться в подынтегральном выражении внутри термов с заголовками "логарифм", "арксинус", "арккосинус", "арктангенс". Каждая содержащая переменную интегрирования тригонометрическая операция должна быть достижима из корня подынтегрального выражения переходами только через символы "плюс", "минус", "умножение", "дробь", "модуль" и через основания степеней. Показатель степени c идентифицируется с натуральной константой, большей единицы. Уровень срабатывания равен 2.

    $\forall_{abcde}(p = a \sin b \sin c/d + e \to a \sin b \sin c/d + e = p)$

    Аналогично предыдущему, но прием активируется не степенью синуса, а произведением синусов.

    (c) Произведение тригонометрических функций, разность либо сумма аргументов которых не зависит от переменной интегрирования. Приемы понижают степень содержащих переменную интегрирования тригонометрических операций, расположенных в знаменателе:

    $\forall_{abcdefg}(d/e(\sin(c+a))^f(\sin(c+b))^g = d\cos(c+b)/e\sin(a-b)(\sin(c+a))^{f-1}(\sin(c+b))^g - d\cos(c+a)/e\sin(a-b)(\sin(c+a))^f(\sin(c+b))^{g-1})$

$\forall_{abcdefg}(d/e(\sin(c{+}a))^f(\cos(c{+}b))^g = d\cos(c{+}a)/e\cos(a{-}b)(\sin(c{+}a))^f(\cos(c{+}b))^{g-1} + d\sin(c+b)/e\cos(a-b)(\sin(c+a))^{f-1}(\cos(c+b))^g)$

$\forall_{abcdefg}(d/e(\sin(a{-}c))^f(\cos(c{+}b))^g = d\cos(a{-}c)/e\cos(a{+}b)(\sin(a{-}c))^f(\cos(c{+}b))^{g-1} - d\sin(c+b)/e\cos(a+b)(\sin(a-c))^{f-1}(\cos(c+b))^g)$

$\forall_{abcdefg}(d/e(\sin(a{-}c))^f(\sin(c{+}b))^g = d\cos(a{-}c)/e\sin(a{+}b)(\sin(a{-}c))^f(\sin(c{+}b))^{g-1} + d\cos(c+b)/e\sin(a+b)(\sin(a-c))^{f-1}(\sin(c+b))^g)$

Дробь входит в условие задачи на преобразование, имеющей цель "нормИнтеграл". Допускаются заголовки "плюс", "умножение", "минус", "модуль" ее надтермов. Переменные f, g идентифицируются с натуральными константами, которые могут вырождаться в единицу. Допускаются также вырожденные нулевые значения a, b и единичное значение e. Переменная интегрирования входит в c и не входит в a, b. Для первых двух приемов проверяется различие выражений a, b. Уровень срабатывания равен 3.

(d) Переход к двойному аргументу. Приемы исключают вхождения в подынтегральное выражение синусов и косинусов заданного аргумента, получая в результате выражение, содержащее только тригонометрические операции с двойным аргументом.

$\forall_{abcd}(a = 2b \to d(\sin c)^a = d(1 - \cos(2c))^b/2^b)$

$\forall_{abcd}(a = 2b \to d(\cos c)^a = d(1 + \cos(2c))^b/2^b)$

Выражение входит в условие задачи на преобразование, имеющей цель "нормИнтеграл". Переменная интегрирования входит в аргумент тригонометрической операции. Показатель степени идентифицируется с натуральной константой. Антецедент, выделенный указателем "программа", проверяет, что эта константа четная, и делит ее на 2. Проверяется, что подынтегральное выражение не содержит тригонометричеких операций с аргументами, отличными от c и $2c$. Проверяется также, что каждое вхождение в подынтегральное выражение тригонометрической операции от c имеет вид $(\sin c)^m$, либо вид $(\cos c)^m$, либо вид $k(\sin c)^n(\cos c)^n$, где k не содержит переменной интегрирования, m делится на 4, а n - четное. Прием блокируется, если подынтегральное выражение имеет единственное вхождение тригонометрической операции, содержащей переменную интегрирования, причем это вхождение достижимо из корня переходами только через "минус", "умножение", "дробь", "модуль" и через основания степеней. Одновременно заменяются все вхождения $d(\sin c)^a$ в условие задачи. Уровень срабатывания равен 2.

$\forall_{abc}(a(\sin c)^b(\cos c)^b = a(\sin(2c))^b/2^b)$

Аналогично предыдущему. Заметим, что фактически фильтры приемов ориентированы на последующее преобразование операций $\sin(2c)$, $\cos(2c)$ в $\cos(4c)$.

(e) Устранение двойного аргумента. Если в подынтегральном выражении уже имеется тригонометрическая операция от аргумента, четным кратным которого является текущий аргумент, то выполняется переход от последнего к половинному аргументу:

$\forall_{ab}(a = b/2 \to \sin b = 2\sin a\cos a)$

Синус входит в условие задачи на преобразование, имеющей цель "нормИнтеграл". Аргумент b содержит переменную интегрирования. Условие задачи имеет вхождение тригонометрической операции от аргумента c, такого, что b/c четно. Требуется, чтобы преобразуемое вхождение было расположено внутри дроби либо произведения, хотя бы два операнда которых содержали переменную интегрирования. Антецедент выделен указателем "идентификатор" и обрабатывает правую часть равенства нормализатором "нормдробь". Уровень срабатывания равен 5.

(f) Переход к одной трети тригонометрического аргумента. Преобразование выполняется, если эта треть уже встречается в условии задачи под другой тригонометрической операцией.

$\forall_{ab}(a = 3b \rightarrow \sin a = 3\sin b - 4(\sin b^3))$

Синус входит в условие задачи на преобразование, имеющей цель "нормИнтеграл". Указатель "контекст(тригаргумент(корень x2))" идентифицирует в условии задачи некоторый тригонометрический аргумент b, после чего выделенный указателем "идентификатор" антецедент сравнивает a и $3b$. Для обработки правой части равенства используется нормализатор "стандплюс". Уровень срабатывания равен 3.

16. Редактирование ответа задачи на вычисление. Приемы применяются в задачах на преобразование, имеющих цель "учетрезультата". Такие задачи возникают при редактировании ответа задач на описание, имеющих цель "известно". Напомним, что обращение к редактированию происходит из блока анализа задачи на описание.

(a) Произведение синусов с одной шестой пи.

$\forall_a(\sin(a + \pi/6)\sin(\pi/6 - a) = (2\cos(2a) - 1)/4)$

Уровень срабатывания равен 3.

(b) Переход к половинному аргументу.

$\forall_a(0 \leq a \ \& \ 0 \leq \pi/2 - a \rightarrow \sqrt{1 - \sin a} = \sqrt{2}\sin(\pi/4 - a/2))$

Уровень срабатывания равен 3.

$\forall_{abcd}(d - 4a = 0 \rightarrow b\sin d/c(\sin a)^2 = 4b\cos(2a)\operatorname{ctg} a/c)$

Уровень срабатывания равен 1.

(c) Приведение подобных членов с синусами.

$\forall_{abc}(a\sin b + c\sin b = (a + c)\sin b)$

Уровень срабатывания равен 1.

17. Одна специальная константная сумма.

$\sin\pi/14 + \cos\pi/7 - \sin 3\pi/14 = 1/2$

Уровень срабатывания равен 2.

Общая стандартизация утверждений

1. Равенство нулю синуса разности углов.

$\forall_{ab}(0 < a \ \& \ 0 \leq \pi - a \ \& \ 0 < b \ \& \ 0 \leq \pi - b \rightarrow \sin(a - b) = 0 \leftrightarrow a - b = 0)$

$\forall_{ab}(0 < a \ \& \ 0 \leq \pi - a \ \& \ 0 < b \ \& \ 0 \leq \pi - b \rightarrow \sin(a/2 - b/2) = 0 \leftrightarrow a - b = 0)$

Попытка применить прием предпринимается, если в посылках задачи имеется неравенство с π, пересекающееся по своим параметрам с аргументом синуса. Уровень срабатывания равен 1.

$\forall_{abABCDEF}(a = \angle(ABC) \ \& \ b = \angle(DEF) \to \sin(a-b) = 0 \leftrightarrow a = b \ \vee \ a = \pi \ \& \ b = 0 \ \vee \ a = 0 \ \& \ b = \pi)$

Прием применяется к содержащей неизвестные посылке задачи на исследование, имеющей цель "известно". Выведенная дизъюнкция сопровождается комментарием "разборслучаев". Уровень срабатывания равен 6.

2. Равенство синуса единице.

   $\forall_a(0 \leq a \ \& \ 0 \leq 2\pi - a \to \sin a = 1 \leftrightarrow a = \pi/2)$

   Прием применяется к посылке задачи на доказательство либо на исследование. Антецеденты обрабатываются проверочными операторами. Проверяется, что a не является недопустимым обозначением угла $\angle(AAB)$. Такие обозначения могут возникать при разборе случаев с отождествлением точек. Тогда из других посылок должно усматриваться противоречие, и блокировка приема предотвращает ненужные действия. Уровень срабатывания равен 1.

3. Невозможность одновременного равенства нулю синуса и косинуса.

   $\forall_a(\cos a = 0 \to \neg(\sin a = 0))$

   Прием имеет заголовок "второйтерм" и заменяет равенство синуса нулю на константу "ложь". Антецедент берется в контексте. Уровень срабатывания равен 1.

4. Исключение синуса в равенстве нулю синуса угла, заключенного между 0 и пи.

   $\forall_a(0 \leq a \ \& \ 0 < \pi - a \to \sin a = 0 \leftrightarrow a = 0)$

   Прием используется только в очень специальных ситуациях - при определении области интегрирования, где он позволил ускорить обработку данных в условиях жесткого ограничения трудоемкости. Антецеденты обрабатываются проверочными операторами. Уровень срабатывания равен 1.

5. Преобразование неравенства для синуса в равенство.

   $\forall_a(0 \leq \sin a - 1 \leftrightarrow \sin a = 1)$

   Уровень срабатывания равен 0.

   $\forall_a(1 \leq \sin a \leftrightarrow \sin a = 1)$

   $\forall_a(\sin a \leq -1 \leftrightarrow \sin a = -1)$

   Уровень срабатывания равен 1.

6. Неотрицательность синуса угла.

   $\forall_{ab}(0 \leq \pi - a \ \& \ 0 \leq a \ \& \ b < 0 \to \neg(\sin a = b))$

   Прием имеет заголовок "второйтерм". Равенство входит в условие задачи на описание, имеющей цель "стандравно". Такие задачи используются для разрешения относительно численных неизвестных подмножеств уравнений задачи на исследование, имеющей цель "известно". Выражение a содержит неизвестные, b - не содержит. Заголовком условия должна служить дизъюнкция, т.е. прием предназначен для устранения разбора случаев. Антецеденты обрабатываются проверочными операторами. Уровень срабатывания равен 2.

7. Невозможность равенства синуса выражению, модуль которого больше единицы.

   $\forall_{abc}(c \leq |b| \ \& \ 0 < c - 1 \rightarrow \neg(\sin a = b))$

   Равенство входит в условие задачи на описание. Выражение a содержит неизвестные, b - не содержит и является неконстантным. Первый антецедент выделен указателем "значения". Он определяет константную нижнюю оценку c модуля b. Второй антецедент обрабатывается проверочным оператором. Прием имеет заголовок "второйтерм", т.е. заменяет равенство на константу "ложь". Введен сравнительно сильный ограничитель трудоемкости. Уровень срабатывания равен 1.

8. Равенство синуса нулю.

   $\forall_{mnk}(n - \text{целое} \ \& \ \neg(k - 1 = 0) \ \& \ m - \text{целое} \rightarrow \sin(n\pi^k/m) = 0 \leftrightarrow n = 0)$

   Выражение k представляет собой десятичную константу. Первый и третий антецеденты обрабатываются проверочными операторами, второй - выделен указателем "программа". Уровень срабатывания равен 1.

Переход к синусу двойного аргумента

$\forall_{abc}(c(\sin a)^b(\cos a)^b = c(\sin(2a))^b/2^b)$

Произведение входит в условие задачи на преобразование. Вне него в условии не встречаются тригонометрические аргументы, целочисленным кратным которым является a (в частности, аргументы, равные a). Задача не имеет цели "нормИнтеграл". Переменные b, c могут вырождаться в единицу. Уровень срабатывания равен 1. На той же теореме основаны еще два приема, применяемые в посылках задач на доказательство либо на исследование. Оба они требуют, чтобы a не содержало неизвестных. Первый из них, срабатывающий на уровне 0, применяется, если посылка не имеет вхождений $\sin a$, $\cos a$ вне рассматриваемого произведения. Второй, срабатывающий на уровне 1, применяется без дополнительных ограничений.

$\forall_a(0 \leq \sin a \cos a \leftrightarrow 0 \leq \sin(2a))$

Неравенство является условием задачи на описание. Должно существовать другое условие, содержащее подвыражение $\sqrt{\sin(2a)}$. Уровень срабатывания равен 0.

$\forall_a(\sqrt{2}\sqrt{\sin a}\sqrt{\cos a} = \sqrt{\sin(2a)})$

$\forall_a(\sqrt{2}\sqrt{-\sin a}\sqrt{-\cos a} = \sqrt{\sin(2a)})$

Произведение входит в условие задачи на описание. Эта задача должна иметь условие с тригонометрической операцией от аргумента $2a$. Уровень срабатывания равен 0.

$\forall_{abxy}(x = 2y \rightarrow a \sin y \cos y + b \sin x = (b + a/2) \sin x)$

Преобразование - разновидность приведения подобных членов. Антецедент выделен указателем "идентификатор". Уровень срабатывания равен 1.

Переход к синусу половинного аргумента

$\forall_{abc}(b = \sin a/2 \ \& \ c = \cos a/2 \to \sin a = 2bc)$

Антецеденты выделены указателем "идентификатор" и обрабатывают правые части равенств нормализаторами общей стандартизации. Прием применяется, если синус является основанием степени сомножителя числителя либо знаменателя дроби, в то время как синус либо косинус половинного аргумента - сомножителем противоположного операнда дроби. Если текущий терм задачи является константным (кроме случая упрощения ответа задач на описание, имеющих цель "известно"), то уровень срабатывания равен 5, иначе он равен 3.

Линейная комбинация синуса и косинуса с коэффициентами $\pm\sqrt{3}, \pm 1$

$\forall_{ab}(\sqrt{3}a\cos b - a\sin b = 2a\cos(b + \pi/6))$

$\forall_{ab}(\sqrt{3}a\cos b + a\sin b = 2a\cos(b - \pi/6))$

$\forall_{ab}(\sqrt{3}a\sin b + a\cos b = 2a\sin(b + \pi/6))$

$\forall_{ab}(\sqrt{3}a\sin b - a\cos b = 2a\sin(b - \pi/6))$

Преобразуемая сумма является одной из частей уравнения, неравенства либо отрицания равенства, представляющего собой условие задачи на описание либо посылку задачи на исследование. Выражение b содержит неизвестные. Каждое вхождение неизвестной в текущий терм задачи расположено внутри рассматриваемой суммы. Уровень срабатывания равен 1.

Синус выражения, содержащего обратные тригонометрические функции

1. Синус арксинуса.

   $\forall_a(\sin \arcsin a = a)$

   Уровень срабатывания равен 0.

2. Синус арккосинуса.

   $\forall_a(0 \leq 1 + a \ \& \ 0 \leq 1 - a \to \sin \arccos a = \sqrt{1 - a^2})$

   Антецеденты обрабатываются проверочными операторами. Для сопровождения по о.д.з. выводится утверждение $0 \leq 1 - a^2$. Если выражение a константное, то уровень срабатывания равен 0, иначе он равен 1. На той же теореме создана еще одна версия приема, применяемая к посылкам задач на исследование. Ее уровень срабатывания равен 0.

3. Синус арктангенса.

   $\forall_a(\sin \operatorname{arctg} a = a/\sqrt{a^2 + 1})$

   $\forall_a(\sin \operatorname{arctg} a = 0 \leftrightarrow a = 0)$

   Уровень срабатывания равен 0.

4. Синус половины арккосинуса.

   $\forall_a(0 \leq 1 + a \ \& \ 0 \leq 1 - a \to \sin(\arccos a/2) = \sqrt{(1 - a)/2})$

Антецеденты обрабатываются проверочными операторами. Созданы две версии приема, в одной из которых правые части неравенств обрабатываются нормализатором "видумножение", а в другой - не обрабатываются. Уровень срабатывания равен 1.

$\forall_a(\sin(\arccos a/2) = \sqrt{(1-a)/2})$

$\forall_a(\sin(\arccos a/2 + \pi/4) = (\sqrt{1-a} + \sqrt{1+a})/2)$

Приемы применяются при редактировании ответа задачи на описание, имеющей цель "известно". Уровень срабатывания равен 1.

5. Синус половины арксинуса.

   $\forall_a(0 \leq a \ \& \ 0 \leq 1-a \rightarrow \sin(\arcsin a/2) = \sqrt{(1-\sqrt{1-a^2})/2})$

   Уровень срабатывания равен 1.

   $\forall_a(\sin(\arcsin a/2) = \sqrt{(1-\sqrt{1-a^2})/2})$

   Прием применяется к посылке задачи на доказательство либо на исследование. Требуется, чтобы a имело вид $b/c\sqrt{d}$ для неконстантного d. Для сопровождения по о.д.з. выводится утверждение $0 \leq 1-a^2$. Уровень срабатывания равен 1.

6. Синус половины арктангенса.

   $\forall_a(0 \leq a \rightarrow \sin(\operatorname{arctg} a/2) = \sqrt{(\sqrt{a^2+1}-1)/2\sqrt{a^2+1}})$

   Выражение находится в условии задачи. Уровни срабатывания равны 2 и 6.

7. Синус одной четвертой арккосинуса.

   $\forall_a(\sin(\arccos a/4) = \sqrt{(1-\sqrt{(1+a)/2})/2})$

   $\forall_a(\sin(\arccos a/4 + \pi/4) = (\sqrt{1+\sqrt{(1+a)/2}} + \sqrt{1-\sqrt{(1+a)/2}})/2)$

   Выражение a константное. Приемы применяются в задаче на исследование, имеющей цель "известно". Уровень срабатывания равен 0.

8. Синус удвоенного арккосинуса.

   $\forall_a(\sin(2\arccos a) = 2a\sqrt{1-a^2})$

   Выражение входит в условие задачи. Уровень срабатывания равен 0.

9. Синус удвоенного арксинуса.

   $\forall_a(\sin(2\arcsin a) = 2a\sqrt{1-a^2})$

   Аналогично предыдущему.

10. Синус удвоенного арктангенса.

    $\forall_a(\sin(2\operatorname{arctg} a) = 2a/(1+a^2))$

    Выражение входит в условие задачи. Уровень срабатывания равен 1.

11. Синус утроенного арксинуса.

    $\forall_a(\sin(3\arcsin a) = 3a - 4a^3)$

    Уровень срабатывания равен 0.

12. Синус утроенного арккосинуса.

    $\forall_a(0 \leq 1+a \ \& \ 0 \leq 1-a \rightarrow \sin(3\arccos a) = (4a^2-1)\sqrt{1-a^2})$

    Для сопровождения по о.д.з. выводится утверждение $0 \leq 1-a^2$. Если выражение a константное, то уровень срабатывания равен 0, иначе - равен 1.

13. Синус учетверенного арктангенса.

    $\forall_a(\sin(4\operatorname{arctg} a) = 4a(1-a^2)/(1+a^2)^2)$

    Преобразуемое выражение входит в условие задачи. Уровень срабатывания равен 1.

14. Синус трех вторых арккосинуса.

    $\forall_a(\sin(3\arccos a/2) = (1+2a)\sqrt{(1-a)/2})$

    Прием применяется в задаче на преобразование, имеющей цель "известны" либо "учетрезультата", т.е. упрощающей ответ задачи на вычисление. Уровень срабатывания равен 1.

15. Синус от линейной комбинации с участием обратной тригонометрической функции.

    $\forall_{ab}(\sin(a+b) = \sin a\cos b + \cos a\sin b)$

    Выражение a, с точностью до знака, имеет один из следующих типов: $k\arcsin A$, $k\arccos A$, $k\operatorname{arctg} A$, $(\arcsin A)/2, (\arccos A)/2, (\operatorname{arctg} A)/2$. Синус входит в условие задачи. Если решается задача на описание, то она должна иметь цель "учетответа" (т.е. выполнять редактирование явного параметрического описания). Либо преобразуемое выражение имеет не более одной свободной переменной, либо решается задача на преобразование, имеющая цель "известны", а число свободных переменных не более 4. Заменяющее выражение упрощается с помощью вспомогательной задачи. Если текущее условие является равенством, то уровень срабатывания равен 1, иначе - равен 3.

    $\forall_{abc}(c = ab + \sqrt{1-b^2}\sqrt{1-a^2} \rightarrow \sin(\arcsin a + \arccos b) = c)$

    Прием применяется в задачах на преобразование, имеющих цель "известны". Уровень срабатывания равен 1.

16. Обращение к вспомогательной задаче для преобразования синуса от выражения, содержащего обратную тригонометрическую функцию.

    $\forall_{ab}(a = \sin b \rightarrow \sin b = a)$

    Синус входит в посылку задачи на исследование, имеющей цель "известно". Выражение b константное, причем в нем встречается хотя бы один из символов "арксинус", "арккосинус", "арктангенс", "арккотангенс". Антецедент выделен указателем "идентификатор" и упрощает синус с помощью вспомогательной задачи на преобразование. Результат a уже не содержит указанных выше символов. Введен средний ограничитель трудоемкости. Уровень срабатывания приема равен 0.

17. Попытка перехода к половинному аргументу.

    $\forall_{abc}(b = \sin a \ \& \ c = \cos a \rightarrow \sin(2a) = 2bc)$

Выражение встречается в условии задачи. Аргумент a имеет вид произведения обратной тригонометрической функции на натуральную константу (возможно, равную единице). Антецеденты выделены указателем "идентификатор" и упрощают правые части равенств с помощью вспомогательных задач на преобразование. Результирующие выражения b, c не содержат символов обратных тригонометрических операций. Введен средний ограничитель трудоемкости. Уровень срабатывания равен 3.

Уравнения

Заметим, что наиболее часто применяемый способ решения тригонометрических уравнений - разложение на множители разности частей уравнения (см. описанный выше прием, отнесенный к символу "умножение"). Другие приемы таковы:

1. Простейшее уравнение для синуса.

   $\forall_{ab}(\sin a = b \leftrightarrow 0 \leq b+1 \ \& \ 0 \leq 1-b \ \& \ (\exists_n(n - \text{целое} \ \& \ a = \arcsin b + 2\pi n) \ \vee \ \exists_n(n - \text{целое} \ \& \ a = -\arcsin b + \pi + 2\pi n)))$

   Равенство является условием задачи на описание, не имеющей целей "известно ..." и "связка ...". Выражение a содержит неизвестные, а выражение b - не содержит. Отсутствует другое уравнение вида $\sin A = 1$, у которого a - целое кратное A. Кроме того, должен быть ложным оператор фильтра "фильтрсерии(a)". Этот оператор реализован на ГЕНОЛОГе и формулирует условия нецелесообразности перехода к параметрическому описанию серии корней тригонометрического уравнения. Он истинен, если выполнено хотя бы одно из следующих требований:

   (a) Уже имеется условие задачи, представляющее собой параметрическое описание для неизвестного выражения, имеющего общую неизвестную с a.

   (b) Выражение a содержит параметр n, для которого имеется условие задачи "целое(n)".

   Преобразованное уравнение сопровождается комментарием "серия", указывающим, что оно теперь представляет собой параметрическое описание значений неизвестной. Фактически, если a не является переменной, такое описание пока не разрешено относительно неизвестных. Оно будет разрешаться с помощью двух средств: сначала - с использованием тех простых приемов, которые могут срабатывать непосредственно под квантором существования, затем - путем перехода к вспомогательной задаче на описание, имеющей цель "учетответа". Последний прием был изложен выше, при рассмотрении реализованных непосредственно на ЛОСе приемов квантора существования.

   Уровень срабатывания приема, решающего уравнение с синусом, равен 2. Для специальных случаев созданы альтернативные приемы решения таких уравнений, применяемые на уровне 1:

   $\forall_a(\sin a = 0 \leftrightarrow \exists_n(n - \text{целое} \ \& \ a = \pi n))$

   $\forall_a(\sin a = 1 \leftrightarrow \exists_n(n - \text{целое} \ \& \ a = \pi(4n+1)/2))$

   $\forall_a(\sin a = -1 \leftrightarrow \exists_n(n - \text{целое} \ \& \ a = \pi(4n-1)/2))$

Приемы аналогичны общему случаю, но в первом и третьем из них отброшен фильтр про другое уравнение вида $\sin A = 1$. По первой теореме создана еще одна версия приема, срабатывающая на уровне 6 без ограничений, накладываемых оператором "фильтрсерии". Для равенства синуса другому синусу либо косинусу (не имеющему неизвестных) тоже созданы специальные приемы:

$\forall_{ab}(\sin a = \cos b \leftrightarrow \exists_n(a = \pi/2 - b + 2\pi n \ \&\ n -$ целое$) \vee \exists_n(a = \pi/2 + b + 2\pi n \ \&\ n -$ целое$))$

$\forall_{ab}(\sin a = -\cos b \leftrightarrow \exists_n(a = -\pi/2 - b + 2\pi n \ \&\ n -$ целое$) \vee \exists_n(a = -\pi/2 + b + 2\pi n \ \&\ n -$ целое$))$

$\forall_{ab}(\sin a = \sin b \leftrightarrow \exists_n(a = b + 2\pi n \ \&\ n -$ целое$) \vee \exists_n(a = \pi - b + 2\pi n \ \&\ n -$ целое$))$

$\forall_{ab}(\sin a = -\sin b \leftrightarrow \exists_n(a = -b + 2\pi n \ \&\ n -$ целое$) \vee \exists_n(a = \pi + b + 2\pi n \ \&\ n -$ целое$))$

Фильтры и указатели для этих специальных случаев - такие же, как для предыдущих. Если задача на описание была создана для решения уравнений, извлеченных из посылок задачи на исследование, имеющей цель "известно" (например, в планиметрии), то она имеет цель "стандравно". В этой ситуации обычно возникают ограничения на интервал изменения неизвестного аргумента синуса, и во избежание рассмотрения серий решений созданы приемы, перехватывающие уравнение у общего приема:

$\forall_{ab}(0 \leq a \ \&\ 0 \leq \pi - 2a \rightarrow \sin a = b \leftrightarrow 0 \leq b + 1 \ \&\ 0 \leq 1 - b \ \&\ a = \arcsin b)$

Уравнение является условием задачи на описание, имеющей цель "стандравно". Выражение a содержит неизвестные, b - не содержит. Антецеденты обрабатываются проверочными операторами, причем введен сильный ограничитель трудоемкости. Уровень срабатывания равен 0.

$\forall_{ab}(0 \leq a \ \&\ 0 \leq \pi - a \rightarrow \sin a = b \leftrightarrow 0 \leq b + 1 \ \&\ 0 \leq 1 - b \ \&\ (a = \arcsin b \ \vee\ a = \pi - \arcsin b))$

$\forall_{ab}(b < 0 \ \&\ 0 \leq a - \pi \ \&\ 0 \leq 2\pi - a \rightarrow \sin a = b \leftrightarrow 0 \leq b + 1 \ \&\ 0 \leq 1 - b \ \&\ (a = 2\pi + \arcsin b \ \vee\ a = \pi - \arcsin b))$

$\forall_a(0 < \pi + a \ \&\ 0 < \pi - a \rightarrow \sin a = 0 \leftrightarrow a = 0)$

Аналогично предыдущему, но ограничитель трудоемкости отсутствует и уровень срабатывания равен 1. Наконец, непосредственно в посылках задачи на исследование, имеющей цель "известно", обрабатывается условие равенства синуса нулю:

$\forall_a(0 \leq a \ \&\ 0 < \pi - a \rightarrow \sin a = 0 \leftrightarrow a = 0)$

Для проверки антецедентов введен сильный ограничитель трудоемкости. Уровень срабатывания равен 2.

2. Однородные тригонометрические уравнения.

   $\forall_{abcdefgh}(f = d - a \ \&\ \neg(f = 0) \ \&\ g = b^2 - 4(d - c)f \ \&\ 0 \leq g \ \&\ h = \sqrt{g} \rightarrow a(\sin e)^2 + b \sin e \cos e + c(\cos e)^2 = d \leftrightarrow \operatorname{tg} e = (b - h)/2f \ \vee\ \operatorname{tg} e = (b + h)/2f)$

   Уравнение является условием задачи на описание. Выражение e содержит неизвестные, выражения a, b, c, d - не содержат. Указатели "подстановка(фикс(0 1 1 3)x3 0)", "подстановка(фикс(0 1 1 1)x1 0)" разрешают обращаться коффицентам a, c в ноль. Однако, хотя бы один из этих коэффициентов должен быть отличен от нуля (иначе получаем простое уравнение, решаемое другими приемами). Первый, третий и пятый антецеденты выделены указателем "идентификатор". При вычислении g используется нормализатор раскрывания скобок "стандплюс", при вычислении h - вспомогательная задача на преобразование. Уровень срабатывания равен 2.

3. Линейная комбинация синуса и косинуса в левой части уравнения.

   $\forall_{abcde}(e = \sqrt{a^2+b^2-c^2} \to a\sin d + b\cos d = c \leftrightarrow \neg(b+c=0)\ \&\ 0 \leq a^2+b^2-c^2\ \&\ (\mathrm{tg}(d/2) = (a+e)/(b+c) \vee \mathrm{tg}(d/2) = (a-e)/(b+c)) \vee b+c=0\ \&\ (\cos(d/2) = 0\ \vee\ a\,\mathrm{tg}(d/2)+b=0))$

   Уравнение является условием задачи на описание либо посылкой задачи на исследование. Выражение d содержит неизвестные, выражения a, b, c - не содержат. Правая часть c ненулевая. Выражение d не содержит целочисленных неизвестных. Антецедент выделен указателем "идентификатор". Подтермы заменяющего терма обрабатываются множеством различных нормализаторов, а дроби для значений тангенса - также вспомогательными задачами на упрощения. Уровень срабатывания равен 2.

   $\forall_{abcd}(0 < d \to a(\sin c)^d + b(\cos c)^d = 0 \leftrightarrow \neg(a=0)\ \&\ (\mathrm{tg}\, c)^d = -b/a\ \vee\ a = 0\ \&\ b\cos c = 0)$

   Уравнение является условием задачи на описание, не имеющей цели "стандравно". Выражение c содержит неизвестные, выражения a, b, d - не содержат. Показатель степени d не равен 2. Уровень срабатывания равен 1.

   $\forall_{abc}(a\cos c + a\sqrt{3}\sin c = b \leftrightarrow 2a\cos(c-\pi/3) = b)$

   $\forall_{abc}(a\cos c - a\sqrt{3}\sin c = b \leftrightarrow 2a\cos(c+\pi/3) = b)$

   $\forall_{abc}(a\cos c - a\sin c = b \leftrightarrow a\sqrt{2}\cos(c+\pi/4) = b)$

   $\forall_{abc}(a\sin c + a\cos c = b \leftrightarrow a\sqrt{2}\sin(c+\pi/4) = b)$

   Уравнение является условием задачи на описание. Выражение c содержит неизвестные, выражения a, b - не содержат. Уровень срабатывания равен 0. Для задач на описание, имеющих цель "стандравно" (решение уравнений, извлекаемых, например, из планиметрических задач), созданы еще два приема:

   $\forall_{abcd}(0 < d\ \&\ \neg(a=0) \to a(\sin c)^d + b(\cos c)^d = 0 \leftrightarrow (\mathrm{tg}\, c)^d = -b/a)$

   $\forall_{abcd}(0 < d\ \&\ \neg(b=0) \to a(\sin c)^d + b(\cos c)^d = 0 \leftrightarrow (\mathrm{ctg}\, c)^d = -a/b)$

   Эти приемы аналогичны приведенному выше приему для общего случая. Они тоже срабатывают на уровне 1.

4. Переход к синусу двойного аргумента, если левая часть уравнения имеет множители - степени синуса и косинуса.

   $\forall_{abcdef}(d = \min(b,c) \to f(\sin a)^b(\cos a)^c = e \leftrightarrow f(\sin(2a))^d(\sin a)^{b-d}(\cos a)^{c-d} = 2^d e)$

   Уравнение является условием задачи на описание. Выражение a содержит неизвестные. Переменные b, c идентифицируются с натуральными константами. Антецедент выделен указателем "программа". Уровень срабатывания равен 3.

5. Равенство плюс-минус единице произведения степеней синусов и косинусов.

   $\forall_{abc}(0 < b \to (\sin a)^b c = -1 \leftrightarrow (\sin a)^b = 1\ \&\ c = -1\ \vee\ (\sin a)^b = -1\ \&\ c = 1)$

$$\forall_{abc}(0 < b \rightarrow (\cos a)^b c = 1 \leftrightarrow (\cos a)^b = 1 \;\&\; c = 1 \;\vee\; (\cos a)^b = -1 \;\&\; c = -1)$$

$$\forall_{abc}(0 < b \rightarrow (\sin a)^b c = 1 \leftrightarrow (\sin a)^b = 1 \;\&\; c = 1 \;\vee\; (\sin a)^b = -1 \;\&\; c = -1)$$

$$\forall_{abc}(0 < b \rightarrow (\cos a)^b c = -1 \leftrightarrow (\cos a)^b = 1 \;\&\; c = -1 \;\vee\; (\cos a)^b = -1 \;\&\; c = 1)$$

Уравнение является условием задачи на описание. Выражение a содержит неизвестные. Каждый сомножитель выражения c является степенью синуса либо косинуса, имеющей положительный показатель. Прием блокируется, если левая часть уравнения представляет собой произведение синуса либо косинуса некоторого аргумента x на синус либо косинус $3x$ - в этом случае будут применяться приемы, преобразующие уравнение к аргументу $2x$. Заменяющая дизъюнкция снабжается комментарием "разборслучаев". Уровень срабатывания равен 1.

6. Усмотрение экстремальных точек. Сумма коэффициентов при произведениях степеней синусов и косинусов сравнивается с известной правой частью:

   $$\forall_{abcde}(e - |a| - |c| = 0 \rightarrow ab + cd = e \leftrightarrow ab = |a| \;\&\; cd = |c|)$$

   $$\forall_{abcde}(e + |a| + |c| = 0 \rightarrow ab + cd = e \leftrightarrow ab = -|a| \;\&\; cd = -|c|)$$

   $$\forall_{abcdefg}(g - |a| - |c| - |e| = 0 \rightarrow ab + cd + ef = g \leftrightarrow ab = |a| \;\&\; cd = |c| \;\&\; ef = |e|)$$

   Уравнение является условием задачи на описание либо посылкой задачи на исследование. Оно содержит синус либо косинус, причем выражения b, d, f идентифицируются как произведения всех сомножителей, имеющих вид степени синуса либо косинуса с неотрицательным показателем. Коэффициенты a, c, e не содержат неизвестных. При наличии цели "известно" прием блокируется. Антецедент выделен указателем "идентификатор", его левая часть обрабатывается нормализаторами общей стандартизации. Уровень срабатывания равен 1.

7. Разбор случаев по целочисленной неизвестной, входящей в аргумент тригонометрической операции.

   $$\forall_{bc}(\exists_e(e \in \{0, \ldots, 2c - 1\} \;\&\; \exists_n(n - \text{целое} \;\&\; b = 2cn + e)))$$

   По этой теореме созданы два приема, один из которых инициируется усмотрением вхождения в условие задачи на описание выражения $\sin(ab\pi/c+d)$, другой - выражения $\cos(ab\pi/c+d)$. Здесь a - целочисленная константа, c - натуральная константа, b - неизвестная, для которой существует условие вида целое(b). Слагаемое d может вырождаться в 0. Приемы имеют заголовок "выводусловия". Указатель "или(фикс(0)фикс(0 2 1))" означает, что внешний квантор существования развертывается в дизъюнкцию утверждений $\exists_n(n-$целое $\&\; b = 2cn+e)$ по всевозможным $c = 0, \ldots, 2n-1$. Эта дизъюнкция и заносится в список условий, сопровождаемая комментариями "разборслучаев" и "серия". Если уже имеется условие с заголовком "или" либо происходит редактирование ответа, то прием блокируется. Срабатывание происходит на уровне 3. Если задача имеет, кроме b, также нецелочисленную неизвестную и не имеет условия с заголовком "существует", то дополнительно разрешается срабатывание на уровне 1.

8. Переход к двойному аргументу в уравнении, определяющем значение квадрата синуса.

   $$\forall_{ax}((\sin x)^2 = a \leftrightarrow \cos(2x) = 1 - 2a)$$

   Уравнение является условием задачи на описание. Выражение x содержит неизвестные, выражение a - не содержит и отлично от 0. Уровень срабатывания равен 0.

9. Равенство единице суммы степеней синуса и косинуса.

   $\forall_{amn}(0 \leq n-2 \ \&\ 0 \leq m-2 \ \&\ (0 < n-2 \ \vee\ 0 < m-2) \rightarrow (\sin x)^n + (\cos x)^m = 1 \leftrightarrow (\sin x)^n = 1 \ \vee\ (\cos x)^m = 1)$

   Уравнение входит в условие задачи на описание. Выражение x содержит неизвестные. Антецеденты обрабатываются проверочными операторами. Уровень срабатывания равен 2.

10. Дизъюнкция условий равенства нулю синуса и косинуса.

    $\forall_{ab}(\sin a = 0 \ \vee\ \cos a = 0 \ \vee\ b \leftrightarrow \sin(2a) = 0 \ \vee\ b)$

    Дизъюнкция является условием задачи на описание. Выражение a содержит неизвестные. Переменная b может идентифицироваться с пустой дизъюнкцией (т.е. с константой "ложь"). Уровень срабатывания равен 0.

11. Отбрасывание в дизъюнкции условия равенства синуса нулю, если в ней же имеется условие равенства синуса двойного аргумента нулю.

    $\forall_{ab}(\sin a = 0 \ \vee\ \sin(2a) = 0 \ \vee\ b \leftrightarrow \sin(2a) = 0 \ \vee\ b)$

    Дизъюнкция является условием задачи на описание. Выражение a содержит неизвестные. Уровень срабатывания равен 0.

12. Кубическое уравнение относительно суммы синуса и косинуса.

    $\forall_{abc}(a(\sin c)^2 \cos c + a \sin c(\cos c)^2 = b \leftrightarrow a(\sin c + \cos c)((\sin c + \cos c)^2 - 1) = 2b)$

    Уравнение является условием задачи на описание. Выражение c содержит неизвестные, выражения a, b - не содержат. Удаление комментария (новыенеизвестные 0) инициирует после срабатывания приема попытку перехода к новой неизвестной $y = \sin c + \cos c$. Уровень срабатывания равен 1.

13. Несовместность условий равенства единице синуса и синуса двойного аргумента.

    $\forall_{a}(\sin a = 1 \rightarrow \neg(\sin(2a) = 1))$

    Уравнение является условием задачи на описание. Заголовок приема - "второйтерм". Уровень срабатывания равен 0.

14. Попытка преобразования линейной комбинации известных синуса и косинуса в правой части уравнения к виду синуса либо косинуса. Предполагается, что в левой части уравнения находится синус с неизвестными:

    $\forall_{abcdx}(d = ab + ac \rightarrow \sin x = ab + ac \leftrightarrow \sin x = d)$

    Уравнение является условием задачи на описание. Выражение x содержит неизвестные, правая часть уравнения неизвестных не содержит. Каждое из выражений b, c, с точностью до знака, представляет собой синус либо косинус. Антецедент выделен указателем "идентификатор" и обрабатывает правую часть нормализатором "видумножение". Результат d , с точностью до знака, представляет собой синус либо косинус. Уровень срабатывания равен 1.

15. Квадратное уравнение относительно линейной комбинации синуса и косинуса.

    $\forall_{ABCDEPQabpqdhr}(C = aq \ \&\ D = bq \ \&\ B = abp \ \&\ \neg(p = 0) \ \&\ 2A + Q - P + pa^2 - pb^2 = 0 \ \&\ d = q^2 + 4pr \ \&\ r = E + (p(a^2 + b^2) - Q - P)/2 \ \&\ h = \sqrt{d} \rightarrow A\cos(2x) +$

$B\sin(2x) + C\sin x + D\cos x + P(\sin x)^2 + Q(\cos x)^2 = E \leftrightarrow (a\sin x + b\cos x = (-q-h)/2p \ \vee \ a\sin x + b\cos x = (-q+h)/2p) \ \& \ 0 \leq d)$

$\forall_{ABCDEPQabpqdhr}(C = aq \ \& \ D = bq \ \& \ B = abp \ \& \ \neg(p = 0) \ \& \ 4A + 2Q - 2P + pa^2 - pb^2 = 0 \ \& \ d = q^2 + 2pr \ \& \ r = E + (p(a^2+b^2) - 2Q - 2P)/4 \ \& \ h = \sqrt{d} \rightarrow A\cos(2x) + B\sin x\cos x + C\sin x + D\cos x + P(\sin x)^2 + Q(\cos x)^2 = E \leftrightarrow (a\sin x + b\cos x = (-q-h)/p \ \vee \ a\sin x + b\cos x = (-q+h)/p) \ \& \ 0 \leq d)$

Антецеденты выражают условия, при которых рассматриваемое уравнение, после перехода к аргументу x, превращается в квадратное уравнение относительно некоторой линейной комбинации синуса и косинуса, а также определяют коэффиценты уравнения и его дискриминант. Все они, кроме четвертого, выделены указателем "идентификатор". Уравнение является условием задачи на описание. Выражение x содержит неизвестные, коэффициенты A, B, C, D, E, P, Q - не содержат. Указатели "подстановка" разрешают обращение в 0 коэффициентов A, P, Q. Первые два антецедента идентифицируют q как произведение общих множителей C, D. Одновременно определяются a, b, и далее третий антецедент позволяет определить p. Уровень срабатывания равен 5.

16. Применение формулы синуса суммы.

    $\forall_{ab}(\sin(a+b) = \sin a\cos b + \cos a\sin b)$

    Синус суммы входит в условие задачи на описание. Переменная b идентифицируется как невырожденная сумма всех неизвестных слагаемых. Выражение a невырождено. В текущем условии встречается тригонометрическая операция с аргументом $\pm b$. Уровень срабатывания равен 6.

17. Применение формулы произведения синусов либо произведения синуса и косинуса для перехода в уравнении к общему неизвестному тригонометрическому аргументу.

    $\forall_{abcd}(c = \cos(a-b) \ \& \ d = \cos(a+b) \rightarrow \sin a\sin b = (c-d)/2)$

    $\forall_{abcd}(c = \sin(a-b) \ \& \ d = \sin(a+b) \rightarrow \sin a\cos b = (c+d)/2)$

    Произведение входит в условие задачи на описание. Выражения a, b содержат неизвестные. Антецеденты выделены указателем "идентификатор". Их правые части обрабатываются нормализаторами общей стандартизации. Одно из выражений c, d неизвестных не содержит. Существует такой тригонометрический аргумент e этих выражений, что вне преобразуемых синуса и косинуса не встречаются тригонометрические аргументы, отличные от $\pm e$. Уровень срабатывания равен 6.

18. Устранение дизъюнкции с суммой синуса и косинуса.

    $\forall_{ax}(\sin x + \cos x = a \ \vee \ \sin x + \cos x = -a \leftrightarrow \sin(2x) = a^2 - 1)$

    Дизъюнкция является условием задачи на описание. Выражение x содержит неизвестные, a - не содержит. Уровень срабатывания равен 2.

19. Переход к квадратному уравнению относительно косинуса.

    $\forall_{abcx}(a(\cos x)^2 + b\sin x = c \leftrightarrow -a(\sin x)^2 + b\sin x = c - a)$

    $\forall_{abcx}(a(\sin x)^2 + b\cos x = c \leftrightarrow -a(\cos x)^2 + b\cos x = c - a)$

Уравнение является условием задачи на описание. Выражение x содержит неизвестные, выражения a, b - не содержат. Указатель "группировка(x2)" определяет идентификацию b путем группировки всех слагаемых с $\sin x$ либо (во втором приеме) с $\cos x$. Уровень срабатывания равен 6.

20. Применение формулы синуса разности для перехода к общему неизвестному тригонометрическому аргументу.

    $\forall_{axy}(x + y = a \to \sin x = \sin a \cos y - \cos a \sin y)$

    Выражение $\sin x$ входит в условие задачи на описание; аргумент x имеет заголовок "плюс" и содержит неизвестные. Указатель "контекст(условие(x2)тригаргумент(x2 x24))" определяет идентификацию в условиях задачи отличного от x тригонометрического аргумента y, тоже содержащего неизвестные. Другие тригонометрические аргументы с неизвестными в текущем условии отсутствуют. Антецедент выделен указателем "идентификатор", его левая часть обрабатывается нормализатором общей стандартизации. Результат a неизвестных не содержит. Уровень срабытывания равен 4.

21. Исключение неизвестных с помощью равенства из контекста.

    $\forall_{abc}(\sin(a + \pi/4) = b \to c \sin a + c \cos a = \sqrt{2}bc)$

    $\forall_{abc}(\cos(a + \pi/4) = b \to c \cos a - c \sin a = \sqrt{2}bc)$

    Преобразуемая сумма либо разность входит в условие задачи на описание. Антецедент берется в контексте. Выражение a содержит неизвестные, выражение b - не содержит. Уровень срабатывания равен 0.

22. Пример ненулевого значения синуса.

    $\forall_x(x = \pi/2 \to \neg(\sin x = 0))$

    Прием имеет заголовок "подборзначений". Отрицание равенства синуса нулю является условием задачи на описание, имеющей цель "пример". Переменная x идентифицируется с неизвестной. Указатель "подборзначений(1)" определяет попытку замены во вспомогательной задаче текущего условия на антецедент. Уровень срабатывания равен 3.

23. Существование решения уравнения.

    $\forall_{ab}(a - \text{число} \ \& \ \neg(b = 0) \to \exists_x(x - \text{число} \ \& \ a = \sin(bx)) \leftrightarrow -1 \leq a \ \& \ a \leq 1)$

    Прием имеет заголовок "связка". Переменная x идентифицируется с несущественной неизвестной задачи на описание. Конъюнктивные члены подкванторного утверждения идентифицируются со всеми условиями задачи, содержащими x. Проверяется, что x не входит в a, b. Прием заменяет все содержащие x условия на конъюнкцию $-1 \leq a \ \& \ a \leq 1$. Уровень срабатывания равен 0.

24. Отрицание равенств с неизвестным синусом.

    (a) Явное разрешение относительно неизвестной.

    $\forall_{abcde}(e\text{–целое} \to \neg(\sin(a\pi/b + ce\pi/d) = 0) \leftrightarrow \exists_f(f \in \{0, \dots, d-1\} \ \& \ \neg(bd|ad + bcf) \ \& \ \exists_n(n - \text{целое} \ \& \ e = dn + f)))$

    Отрицание равенства является условием задачи на описание. Переменная e идентифицируется с неизвестной, выражения a, b, d суть натуральные константы, выражение c - целочисленная константа. Отсутствуют условия с заголовком "или". Указатель "или(фикс(0 2)фикс(0 2 2 1)фикс(0 2 2 2))" определяет развертку заменяющего квантора существования в дизъюнкцию утверждений $\exists_n(n-$целое $\& \ e = dn+f)$, которая берется по всем значениям $f \in \{0, \ldots, d-1\}$, удовлетворяющим ограничению $\neg(bd|ad+bcf)$. Эта дизъюнкция сопровождается комментариями "серия" (помечает параметрическое описание значений неизвестной) и "разборслучаев" (форсирует разбор случаев). Уровень срабатывания равен 2.

$\forall_{abc}(c -$ целое $\rightarrow \neg(\sin(ac\pi/b) = 0) \leftrightarrow \exists_d(d \in \{0, \ldots, b-1\} \ \& \ \neg(d = 0) \ \& \ \exists_n(n -$ целое $\& \ c = bn + d)))$

Аналогично предыдущему. С неизвестной идентифицируется c; выражения a, b суть натуральные константы.

(b) Группировка отрицаний равенства нулю синуса и косинуса.

$\forall_a(\neg(\sin a = 0) \ \& \ \neg(\cos a = 0) \leftrightarrow \neg(\sin(2a) = 0))$

Группировка выполняется, чтобы на этапе проверки обрабатывать не два утверждения, а одно. Требуется, чтобы группируемые утверждения не использовались для сопровождения по о.д.з. Прием имеет заголовок "заменаусловия(второйтерм)", т.е. оба отрицания равенств суть условия задачи на описание. Выражение a содержит неизвестные. Существует хотя бы одно условие, имеющее вид равенства, в одной части которого находится неизвестный синус, косинус либо тангенс, а другая часть известна. Ни одно из таких уравнений не имеет одной из своих частей $\sin a$ либо $\cos a$. Уровень срабатывания равен 1.

(c) Разгруппировка условия равенства нулю синуса двойного угла. Преобразование обратно предыдущему, т.е. теорема приема та же самая, но заголовок - "первыйтерм". Отрицание равенства является условием задачи на описание, не используемым для сопровождения по о.д.з. Существует условие вида $\sin a = b$ либо $\cos a = b$ с известным b. Уровень срабатывания приема равен 1.

(d) Исключение неизвестной из отрицания равенства синуса нулю.

$\forall_{ab}(\cos(2a) = b \rightarrow \neg(\sin a = 0) \leftrightarrow \neg(1 - b = 0))$

Отрицание равенства является условием задачи на описание и не используется для сопровождения по о.д.з. Антецедент берется в контексте (т.е. является другим условием). Выражение a содержит неизвестные, выражение b - не содержит. Уровень срабатывания равен 1.

(e) Исключение избыточного условия равенства синуса нулю.

$\forall_{ab}(b = 2a \ \& \ \neg(\sin b = 0) \rightarrow \neg(\sin a = 0))$

$\forall_{ab}(b = 4a \ \& \ \neg(\sin b = 0) \rightarrow \neg(\sin a = 0))$

Приемы имеют заголовок "второйтерм". Отрицание равенства является условием задачи на описание, не используемым для сопровождения по о.д.з. Прием заменяет его на логическую константу "истина". Второй антецедент берется в контексте, первый - выделен указателем "идентификатор". Для сравнения тригонометрических аргументов его правая часть обрабатывается нормализатором раскрывания скобок "стандплюс". Требуется, чтобы отсутствовало условие, выражающее отличие от нуля синуса $a/2$. Уровень срабатывания равен 0.

$\forall_{abc}(b = 4a \ \& \ \sin b < c \ \& \ c \leq 0 \to \neg(\sin a = 0))$

$\forall_{abc}(b = 4a \ \& \ c < \sin b \ \& \ 0 \leq c \to \neg(\sin a = 0))$

Аналогично предыдущему. Второй антецедент берется в контексте, третий - обрабатывается проверочным оператором.

(f) Группировка отрицаний равенств нулю синусов суммы и разности.

$\forall_{a}(\neg(1 - \sin a = 0) \ \& \ \neg(1 + \sin a = 0) \leftrightarrow \neg(\cos a = 0))$

Прием имеет заголовок "заменаусловия(второйтерм)". Отрицания равенств представляют собой условия задачи на описание, не используемые для сопровождения по о.д.з. Выражение a содержит неизвестные. Уровень срабатывания равен 1.

(g) Группировка отрицаний равенств нулю суммы и разности синуса и косинуса.

$\forall_{ab}(\neg(\cos a + \sin a = 0) \ \& \ \neg(b\cos a - b\sin a = 0) \leftrightarrow \neg(\cos(2a) = 0))$

Множитель b введен для того, чтобы охватить случай $\sin a - \cos a = 0$. Фактически, ввиду предварительного сокращения, этот множитель обычно будет равен ±1. Как и выше, отрицания равенств суть условия задачи на описание, не используемые для сопровождения по о.д.з. Фильтр "десчисло(x2)" допускает идентификацию b лишь с десятичными константами. Указатель "заменазнака(минус x2)" предоставляет возможность одновременного изменения знака у членов разности путем отнесения знака "минус" к коэффициенту b. Уровень срабатывания равен 1.

(h) Учет отрицания равенства нулю синуса либо косинуса, если синус двойного аргумента равен нулю.

$\forall_{a}(\neg(\sin a = 0) \to \sin(2a) = 0 \leftrightarrow \cos a = 0)$

$\forall_{a}(\neg(\cos a = 0) \to \sin(2a) = 0 \leftrightarrow \sin a = 0)$

Уравнение является условием задачи на описание. Антецедент берется в контексте (т.е. является другим условием задачи). Указатель "удалениеусловия(не(равно(синус(x1)0)))" (во втором приеме - аналогичный указатель для косинуса) обеспечивает отбрасывание идентифицированного с антецедентом условия, если оно не используется для сопровождения по о.д.з. Уровень срабатывания равен 0.

(i) Разгруппировка отрицания равенства единице разности синуса и косинуса.

$\forall_{ab}(b = a/2 \to \neg(1 - \cos a + \sin a = 0) \leftrightarrow \neg(\sin b = 0) \ \& \ \neg(\sin(b + \pi/4) = 0))$

Отрицание равенства является условием задачи на описание, не используемым для сопровождения по о.д.з. Антецедент выделен указателем "идентификатор", его правая часть обрабатывается нормализатором раскрывания скобок "стандплюс". Таким образом определяется тригонометрический аргумент b. Проверяется существование условия задачи, в которое входит этот тригонометрический аргумент. Проверяется также существование уравнения, одна часть которого есть тригонометрическая операция с неизвестными, а другая часть известна. Уровень срабатывания равен 1.

25. Рассмотрение синуса от обеих частей уравнения с обратными тригонометрическими функциями.

$\forall_{abcd}(a = \sin b \ \& \ c = \sin d \ \& \ b + d = 0 \to a + c = 0)$

Прием имеет заголовок "выводусловия". Третий антецедент идентифицируется с условием задачи на описание. Слагаемое b имеет своим множителем обратную тригонометрическую операцию от выражения с неизвестными. Ни одно из слагаемых суммы $b + d$ не имеет своим множителем выражение с неизвестными, не содержащее обратных тригонометрических функций. Первый и второй антецеденты выделены указателем "идентификатор". Их правые части упрощаются вспомогательными задачами на преобразование. Для блокировки повторных применений приема используется комментарий "арктангенс". При решении дифференциальных уравнений прием не применяется. Уровень срабатывания равен 3.

$\forall_{abc}(a = \sin b \ \& \ b = c \to a = \sin c)$

Аналогично предыдущему, но правая часть исходного уравнения ненулевая.

26. Системы уравнений.

    (a) Определение синусов либо косинусов суммы и разности из двух уравнений для произведений синусов и косинусов.

    $\forall_{abcd}(\sin a \cos b = c \ \& \ \cos a \sin b = d \leftrightarrow \sin(a+b) = c+d \ \& \ \sin(a-b) = c-d)$

    $\forall_{abcd}(\sin a \sin b = c \ \& \ \cos a \cos b = d \leftrightarrow \cos(a+b) = d-c \ \& \ \cos(a-b) = c+d)$

    Приемы имеют заголовок "заменаусловия(второйтерм)". Уравнения являются условиями задачи на описание, имеющей более одной неизвестной. Выражения a, b содержат неизвестные, c, d - не содержат. Уровни срабатывания равны 2 и 4.

    (b) Линейная комбинация квадратов тригонометрических уравнений для исключения неизвестной по формуле суммы квадратов синуса и косинуса.

    $\forall_{abcdefghij}(i = gd \ \& \ h = ga \ \& \ h \sin j + b = c \ \& \ i \cos j + e = f \to a^2g^2d^2 - a^2(f-e)^2 - d^2(c-b)^2 = 0)$

    Прием имеет заголовок "вывод". Третий и четвертый антецеденты - уравнения, расположенные в списке посылок задачи на исследование. Эта задача может и не иметь цели "известно", т.е. сопровождать обычную задачу на решение системы уравнений. Первый и второй антецеденты выделены указателем "идентификатор". Они определяют общий множитель g коэффициентов h, i. Выражение j содержит такую неизвестную, которая не встречается в выражениях h, i, b, e. Если задача имеет цель "известно", то выражения i, h, b, e, c, f должны содержать ровно один неизвестный числовой атом. Для блокировки повторного вывода используется комментарий (нормсинус ...), сопровождающий каждое из исходных уравнений. Уровень срабатывания равен 2. Создана еще одна версия приема, срабатывающая на уровне 12. В ней отсутствует ограничение на число неизвестных числовых атомов выражений i, h, b, e, c, f, но зато требуется, чтобы эти выражения не имели тригонометрических операций с неизвестными.

    (c) Замена двух тригонометрических уравнений на эквивалентные им линейные комбинации для выделения синусов либо косинусов от суммы и разности одинаковых неизвестных выражений.

    $\forall_{abcdefghijkpq}(h = ag \ \& \ i = dg \ \& \ \neg(h = 0) \ \& \ \neg(i = 0) \ \& \ p = gad\cos(j+k) + ae - db - fa + dc \ \& \ q = gad\cos(j-k) + ae + db - fa - dc \to h \sin j \sin k + b = c \ \& \ i \cos j \cos k + e = f \leftrightarrow p = 0 \ \& \ q = 0)$

$\forall_{abcdefghijkpq}(h = ag \ \& \ i = dg \ \& \ \neg(h = 0) \ \& \ \neg(i = 0) \ \& \ p = gad\sin(k - j) + ae - db - fa + dc \ \& \ q = gad\sin(j + k) + ae + db - fa - dc \rightarrow h\sin j\cos k + b = c \ \& \ i\cos j\sin k + e = f \leftrightarrow p = 0 \ \& \ q = 0)$

Приемы имеют заголовок "заменаусловия(второйтерм)". Уравнения суть условия задачи на описание, имеющей не менее двух неизвестных. Выражения j, k содержат неизвестные. Первый и второй антецеденты выделены указателем "идентификатор" и определяют общий множитель g коэффициентов h, i. Третий и четвертый антецеденты обрабатываются проверочными операторами. Два последних антецедента тоже выделены указателем "идентификатор". Они определяют ненулевые части p, q новых уравнений, предпринимая попытку разложить их на множители. Для определения целесообразности преобразования используется тот же оператор фильтра "фильтрмножителей", который ранее использовался для оценки целесообразности перехода к новому уравнению после попытки разложения на множители разности его частей (см. приемы символа "умножение"). При этом первое уравнение сравнивается с p, а второе - с q. Уровень срабатывания равен 3.

(d) Перемножение однородных тригонометрических уравнений для разделения неизвестных пуем перехода от произведения тригонометрических операций к сумме.

$\forall_{abcd}(a + b = 0 \ \& \ c + d = 0 \rightarrow ac - bd = 0)$

Прием имеет заголовок "вывод". Антецеденты идентифицируются с уравнениями - посылками задачи на исследование, число неизвестных которой больше единицы. Слагаемое a имеет вид произведения синуса либо косинуса тригонометрического аргумента $x + y$ на известный коэффициент, слагаемое c - вид произведения синуса либо косинуса тригонометрического аргумента $x - y$ на известный коэффициент. Выражения x, y содержат неизвестные, но не имеют общей неизвестной. Левая часть выводимого уравнения обрабатывается нормализатором раскрывания скобок "стандплюс". Комментарий "видумножение", которым снабжается обращение к этому нормализатору, обеспечивает преобразование произведений тригонометрических операций в суммы. Для блокировки повторного вывода используется комментарий (косеканс ...). Уровень обращения равен 1.

(e) Вывод линейной комбинации уравнений с целью получения тригонометрической функции от суммы или разности аргументов.

$\forall_{abcdefghijk}(h = ag \ \& \ i = dg \ \& \ h\sin j\sin k + b = c \ \& \ i\cos j\cos k + e = f \rightarrow gad\cos(j + k) + ae - db = fa - dc)$

$\forall_{abcdefghijk}(i = ck \ \& \ j = fk \ \& \ i\sin a\cos b + d = e \ \& \ j\cos a\sin b + g = h \rightarrow kcf\sin(a + b) + df + gc = ef + hc)$

$\forall_{abcdefghijk}(g = ai \ \& \ h = di \ \& \ g\sin j\sin k + b = c \ \& \ h\cos j\cos k + e = f \rightarrow iad\cos(j - k) + bd + ea = cd + fa)$

$\forall_{abcdefghijk}(h = ba \ \& \ i = ea \ \& \ h\sin j\cos k + c = d \ \& \ i\cos j\sin k + f = g \rightarrow abe\sin(j - k) + ce - fb = de - gb)$

Третий и четвертый антецеденты идентифицируются с посылками задачи на исследование, не имеющей цели "известно". Тригонометрические аргументы, выделенные в теоремах приемов, содержат неизвестные. Первый и второй антецеденты выделены указателем "идентификатор" и используются для выделения общего множителя коэффициентов при тригонометрических произведениях. Этот общий множитель может вырождаться в единицу. Уровни срабатывания равны 3.

(f) Перенесение во внешнюю задачу на описание равенства, определяющего значение синуса с неизвестными. Прием имеет теорему вида "замещениеусловий(равно(синус(a)b))". Заголовок приема - "замещениеусловий". Уравнение $\sin a = b$ является посылкой задачи на исследование, представляющей собой блок анализа некоторой внешней задачи на описание Z, не имеющей цели "известно". Выражение a содержит неизвестные, b - не содержит. Либо a имеет более одной неизвестной, либо задача Z не имеет условия, представляющего собой равенство неизвестной тригонометрической операции известному выражению и пересекающегося с a по своим неизвестным. Уравнение $\sin a = b$ переносится в список условий задачи Z, а рассмотрение блока анализа обрывается. Уровень срабатывания равен 1.

27. Использование нормализатора "тригнеизв". Чтобы уменьшить число различных тригонометрических неизвестных аргументов в уравнении, создан нормализатор "тригнеизв". К нему обращается следующий прием:

$\forall_{abc}(c = (a = b) \to (a = b) = c)$

Заголовок приема - "второйтерм". Уравнение $a = b$ является условием задачи на описание, имеющим тригонометрические аргументы $x + a$, $x + b$ либо $x+a$,$b-x$, где a, b не содержат неизвестных, x - содержит. Допускаются другие неизвестные тригонометрические аргументы. Антецедент выделен указателем "идентификатор" и преобразует уравнение нормализатором "тригнеизв". Проверяется, что результат c имеет не более одного неизвестного тригонометрического аргумента (быть может, входящего в уравнение несколько раз). Уровень срабатывания равен 5. В нормализатор "тригнеизв"вошли следующие приемы:

(a) Преобразование произведения тригонометрических операций в сумму.

$\forall_{abcd}(c = \cos(a - b)\ \&\ d = \cos(a + b) \to \sin a \sin b = (c - d)/2)$

$\forall_{abcd}(c = \sin(a - b)\ \&\ d = \sin(a + b) \to \sin a \cos b = (c + d)/2)$

$\forall_{abcd}(c = \cos(a - b)\ \&\ d = \cos(a + b) \to \cos a \cos b = (c + d)/2)$

Выражения a, b содержат неизвестные. Антецеденты выделены указателем "идентификатор", их правые части обрабатываются нормализаторами общей стандартизации. Либо c, либо d не содержит неизвестных.

(b) Преобразование суммы тригонометрических функций в произведение.

$\forall_{abcde}(d = \cos((a + b)/2)\ \&\ e = \sin((a - b)/2) \to c \sin a - c \sin b = 2cde)$

$\forall_{abcde}(d = \cos((a - b)/2)\ \&\ e = \sin((a + b)/2) \to c \sin a + c \sin b = 2cde)$

$\forall_{abcde}(d = \cos((a + b)/2)\ \&\ e = \cos((a - b)/2) \to c \cos a + c \cos b = 2cde)$

$\forall_{abcde}(d = \sin((a + b)/2)\ \&\ e = \sin((a - b)/2) \to c \cos a - c \cos b = -2cde)$

Выражения a, b содержат неизвестные. Антецеденты выделены указателем "идентификатор". Одно из выражений d, e неизвестных не содержит.

(c) Переход к косинусу половинного аргумента.

$\forall_{ab}(a - 2b = 0 \to \cos a = 2(\cos b)^2 - 1)$

Указатель "контекст(тригаргумент(x2))" определяет идентификацию второго аргумента b внутри преобразуемого терма. Оба выражения a, b содержат неизвестные. Антецедент, выделенный указателем "идентификатор", проверяет, что a равно $2b$.

Неравенства

1. Решение простейших нестрогих неравенств с синусом.

   $\forall_{ab}(\sin b \leq a \leftrightarrow 0 \leq a-1 \ \vee \ a-1 \leq 0 \ \& \ 0 \leq 1+a \ \& \ \exists_n(n - \text{целое} \ \& \ 2\pi n - \pi - \arcsin a \leq b \ \& \ b \leq 2\pi n + \arcsin a))$

   $\forall_{ab}(a \leq \sin b \leftrightarrow 1+a \leq 0 \vee 0 \leq 1-a \ \& \ 0 \leq 1+a \ \& \ \exists_n(n - \text{целое} \ \& \ 2\pi n + \arcsin a \leq b \ \& \ b \leq 2\pi n + \pi - \arcsin a))$

   Неравенство является условием задачи на описание. Выражение b содержит неизвестные, a - не содержит. Отсутствует другое неравенство для $\sin b$, имеющее то же направление (строгое либо нестрогое). Общее число условий - неравенств для $\sin b$ - не превосходит 2. Отсутствуют уравнения, причем преобразуемое неравенство не используется для сопровождения по о.д.з. Кроме этих ограничений, проверяется ложность оператора фильтра "фильтрпромежутков". Этот оператор усматривает нецелесообразность перехода от неравенства для тригонометрического аргумента b к параметрическому описанию серии промежутков. Он истинен при выполнении любого из следующих требований:

   (a) В условиях задачи имеется тригонометрическое неравенство, не разрешенное явно относительно тригонометрической функции неизвестного аргумента.

   (b) Существует условие - параметрическое описание для серии точек.

   (c) Существует условие с заголовком "или".

   Выводимое приемом параметрическое описание серии промежутков сопровождается комментарием "серия". Уровни срабатывания равны 2 и 6.

   Если неизвестная пробегает промежуток $[-\pi, \pi]$, используются следующие два специальных приема:

   $\forall_{ab}(0 \leq \pi + b \ \& \ 0 \leq \pi - b \rightarrow a \leq \sin b \leftrightarrow a+1 < 0 \ \vee \ 0 \leq 1+a \ \& \ 0 \leq 1-a \ \& \ (\arcsin a \leq b \ \& \ b \leq \pi - \arcsin a \ \vee \ b \leq -\pi - \arcsin a))$

   $\forall_{ab}(0 \leq \pi + b \ \& \ 0 \leq \pi - b \rightarrow \sin b \leq a \leftrightarrow 0 < a-1 \ \vee \ 0 \leq 1+a \ \& \ 0 \leq 1-a \ \& \ (-\pi - \arcsin a \leq b \ \& \ b \leq \arcsin a \ \vee \ \pi - \arcsin a \leq b))$

   Как и выше, неравенство является условием задачи на описание. Фильтр "цель(неравенства)" введен потому, что приемы пока понадобились лишь при определении областей интегрирования. Требуется, чтобы задача не имела других неравенств, содержащих тригонометрический аргумент b. Уровень срабатывания равен 1.

2. Решение простейших строгих неравенств с синусом.

   $\forall_{ab}(\sin b < a \leftrightarrow 0 < a-1 \ \vee \ a-1 \leq 0 \ \& \ 0 \leq 1+a \ \& \ \exists_n(n - \text{целое} \ \& \ 2\pi n - \pi - \arcsin a < b \ \& \ b < 2\pi n + \arcsin a))$

   $\forall_{ab}(a < \sin b \leftrightarrow 1+a < 0 \vee 0 \leq 1-a \ \& \ 0 \leq 1+a \ \& \ \exists_n(n - \text{целое} \ \& \ 2\pi n + \arcsin a < b \ \& \ b < 2\pi n + \pi - \arcsin a))$

Аналогично случаю нестрогих неравенств.

3. Решение двусторонних неравенств с синусами.

   $\forall_{abc}(0 \le 1-a \mathbin{\&} 0 \le 1+a \mathbin{\&} 0 \le 1-b \mathbin{\&} 0 \le 1+b \to a \le \sin c \mathbin{\&} \sin c \le b \leftrightarrow 0 \le b-a \mathbin{\&} (\exists_n(n - \text{целое} \mathbin{\&} 2\pi n + \arcsin a \le c \mathbin{\&} c \le 2\pi n + \arcsin b) \vee \exists_n(n - \text{целое} \mathbin{\&} 2\pi n - \pi - \arcsin b \le c \mathbin{\&} c \le 2\pi n - \pi - \arcsin a)))$

   Прием имеет заголовок "заменаусловия(второйтерм)". Оба неравенства являются условиями задачи на описание, не используемыми для сопровождения по о.д.з. Задача не имеет уравнений. Выражение c содержит неизвестные, выражения a, b - не содержат. Отсутствуют другие неравенства для $\sin c$. Оператор фильтра "фильтрпромежутков" имеет значение "ложь". Выводимое утверждение сопровождается комментарием "серия". Уровни срабатывания равны 1 и 5. Для рассмотрения строгих неравенств введен отдельный прием, теорема которого почти такая же, как выше:

   $\forall_{abc}(0 \le 1-a \mathbin{\&} 0 \le 1+a \mathbin{\&} 0 \le 1-b \mathbin{\&} 0 \le 1+b \to a \le \sin c \mathbin{\&} \sin c \le b \leftrightarrow 0 < b-a \mathbin{\&} (\exists_n(n - \text{целое} \mathbin{\&} 2\pi n + \arcsin a \le c \mathbin{\&} c \le 2\pi n + \arcsin b) \vee \exists_n(n - \text{целое} \mathbin{\&} 2\pi n - \pi - \arcsin b \le c \mathbin{\&} c \le 2\pi n - \pi - \arcsin a)))$

   Указатели "альтернатива(...)" разрешают в каждом из преобразуемых неравенств выбирать вместо знака "меньшеилиравно" знак "меньше". Соответственно корректируются знаки в заменяющем утверждении. Введен фильтр, требующий, чтобы хотя бы одно из неравенств было строгим. Этим и объясняется замена неравенства $0 \le b-a$ на $0 < b-a$. Дополнительно требуется, чтобы задача не имела условия вида $\neg(\sin c = d)$, так как такое условие будет преобразовано к виду двух неравенств.

4. Линейная комбинация синуса и косинуса в неизвестной части неравенства.

   $\forall_{abc}(a\cos c - a\sin c \le b \leftrightarrow \sqrt{2}a\cos(c+\pi/4) \le b)$

   $\forall_{abc}(a\sin c + a\cos c \le b \leftrightarrow \sqrt{2}a\sin(c+\pi/4) \le b)$

   Неравенство является условием задачи на описание, не используемым для сопровождения по о.д.з. Задача не имеет уравнений. Выражение c содержит неизвестные, выражения a, b - не содержат. Указатели "дробь" и "альтернатива" разрешают перестановку частей неравенства и переход к строгому неравенству. Уровень срабатывания равен 1.

   $\forall_{abcp}(0 < a \mathbin{\&} p - \text{rational} \mathbin{\&} \neg(\text{числитель}(p) - \text{even}) \mathbin{\&} \neg(\text{знаменатель}(p) - \text{even}) \to 0 \le a(\sin c)^p + b(\cos c)^p \leftrightarrow \exists_n(n - \text{целое} \mathbin{\&} 2\pi n + \operatorname{arctg}(-(b/a)^{1/p}) \le c \mathbin{\&} c \le 2\pi n + \pi + \operatorname{arctg}(-(b/a)^{1/p})))$

   $\forall_{abcp}(0 < a \mathbin{\&} p - \text{rational} \mathbin{\&} \neg(\text{числитель}(p) - \text{even}) \mathbin{\&} \neg(\text{знаменатель}(p) - \text{even}) \to a(\sin c)^p + b(\cos c)^p \le 0 \leftrightarrow \exists_n(n - \text{целое} \mathbin{\&} c \le 2\pi n + \operatorname{arctg}(-(b/a)^{1/p}) \mathbin{\&} 2\pi n + \pi + \operatorname{arctg}(-(b/a)^{1/p}) \le c))$

   $\forall_{abcp}(a < 0 \mathbin{\&} 0 \le b \mathbin{\&} p - \text{rational} \mathbin{\&} \text{числитель}(p) - \text{even} \mathbin{\&} 0 \le \sin c \mathbin{\&} 0 \le \cos c \to 0 \le a(\sin c)^p + b(\cos c)^p \leftrightarrow \exists_n(n - \text{целое} \mathbin{\&} 2\pi n + \operatorname{arctg}((-b/a)^{1/p}) \le c \mathbin{\&} c \le 2\pi n + \pi + \operatorname{arctg}((-b/a)^{1/p})))$

   Аналогично предыдущему. Указатели "альтернатива" разрешают рассмотрение строгих неравенств. Показатель степени p - константное выражение. Выводимое утверждение сопровождается комментарием "серия". Уровни срабатывания равны 1 и 2.

5. Сравнение суммы степеней синуса и косинуса с единицей.

   $\forall_{abcd}(0 < 2 - a \ \&\ 0 < 2 - b \ \&\ 0 \leq \sin d \ \&\ 0 \leq \cos d \rightarrow 1 < (\sin d)^a + (\cos d)^b \leftrightarrow 0 < \sin d \ \&\ 0 < \cos d)$

   Антецеденты обрабатываются проверочными операторами. Неравенство расположено в условии задачи. Уровень срабатывания равен 2.

6. Разбиение промежутка значений синуса на два подпромежутка для учета отрицания равенства.

   $\forall_{abcde}(a+d/e < 0 \ \&\ 0 < b+d/e \ \&\ \neg(e=0) \rightarrow a \leq \sin c \ \&\ \sin c \leq b \ \&\ \neg(e \sin c + d = 0) \leftrightarrow a \leq \sin c \ \&\ \sin c < -d/e \ \vee\ -d/e < \sin c \ \&\ \sin c \leq b)$

   Прием имеет заголовок "заменаусловия(второйтерм)". Неравенства и отрицание равенства суть условия задачи на описание, не имеющей уравнений. Выражение c содержит неизвестные, выражения a, b, e, d - не содержат. Антецеденты проверяют, что "запрещенная" для синуса точка $-d/e$ попадает на интервал (a, b). Они обрабатываются проверочными операторами. Проверяется также, что оставшиеся условия задачи не содержат тригонометрических операций с неизвестными. Указатели "альтернатива" обеспечивают применение приема, если вместо нестрогих неравенств берутся строгие. Заменяющее утверждение сопровождается комментарием "разборслучаев". Уровень срабатывания равен 0.

7. Преобразование отрицания равенства с неизвестным синусом в дизъюнкцию двух неравенств.

   $\forall_{ax}(\neg(\sin x = a) \leftrightarrow \sin x < a \ \vee\ a < \sin x)$

   Отрицание равенства является условием задачи на описание, не имеющей уравнений. Выражение x содержит неизвестные, a - не содержит. Задача имеет неравенство, одна часть которого не содержит неизвестных, а другая - является либо неизвестной, либо тригонометрической операцией с неизвестными. Отсутствуют дизъюнктивные условия с неизвестными. При определении области интегрирования прием блокируется. Выводимая дизъюнкция сопровождается комментарием "разборслучаев". Уровень срабатывания равен 1.

   $\forall_{abx}(\neg(a = 0) \rightarrow \neg(a \sin x + b = 0) \leftrightarrow \sin x < -b/a \ \vee\ -b/a < \sin x)$

   Аналогично предыдущему.

8. Неравенства с произведением тригонометрических операций от суммы и разности заданных аргументов.

   $\forall_{ab}(0 < \sin(a+b)\sin(a-b) \leftrightarrow \cos(2a) < \cos(2b))$

   $\forall_{ab}(0 < \sin(a+b)\cos(a-b) \leftrightarrow \sin(2a) < \sin(2b))$

   $\forall_{ab}(0 < \cos(a+b)\sin(a-b) \leftrightarrow \sin(2b) < \sin(2a))$

   Неравенство является условием задачи на описание, не имеющей уравнений. Оно не используется для сопровождения по о.д.з. Одно из выражений a, b содержит неизвестные, а другое - не содержит. Указатели "дробь", "альтернатива" допускают переход к нестрогим неравенствам и перестановку частей неравенства. Уровень срабатывания равен 1.

9. Усмотрение неравенства, зависящего от суммы синуса и косинуса.

   $\forall_{afghxy}((y - \text{число}\ \&\ a \leq f(y)) = g(y) \to a \leq f(\sin h(x) + \cos h(x)) \leftrightarrow g(\sin h(x) + \cos h(x)))$

   $\forall_{afghxy}((y - \text{число}\ \&\ a < f(y)) = g(y) \to a < f(\sin h(x) + \cos h(x)) \leftrightarrow g(\sin h(x) + \cos h(x)))$

   Неравенство является условием задачи на описание, не имеющей уравнений и имеющей единственную неизвестную. Левая часть a не содержит неизвестной. Указатель "контекст(неизвестная(x23) позиция(x3 корень) вид(x3 плюс(умножение(x4 синус(значение(x8 x23)))умножение(x4 косинус(значение(x8 x23)))x5)) единица(1 x4)единица(0 x5) заменазнака(минус x4))" определяет усмотрение в этом неравенстве подвыражения вида $d \sin h(x) + d \cos h(x) + e$. Здесь x - неизвестная, $h(x)$ - функциональная переменная, идентифицируемая с произвольным выражением. Допускаются вырожденные нулевое значение для e и единичное для d. Далее указатель "новаргумент(x6 x23 извлечение)" определяет проверку представимости правой части неравенства в таком виде, что переменная x встречается только внутри подвыражений $\sin h(x)+\cos h(x)$. Здесь используется нормализатор "извлечение", выполняющий необходимые для данного представления преобразования. Переменная f - функциональная. Антецедент выделен указателем "идентификатор". Его левая часть получена из исходного неравенства переобозначением подвыражения $\sin h(x) + \cos h(x)$ на новую переменную y и добавлением условия "y – число". Эта часть разрешается относительно y с помощью вспомогательной задачи на описание. Заменяющее утверждение получается обратной подстановкой в результат выражения $\sin h(x) + \cos h(x)$ вместо y. Если $g(y)$ линейно по y, то прием блокируется, так как в этом случае неравенство решается более простыми средствами. Введен слабый ограничитель трудоемкости. Указатель "дробь" разрешает перестановку частей неравенства. Уровень срабатывания равен 2.

Задачи на исследование, имеющие цель "известно"

При решении задач на исследование, имеющих цель "известно", многие из обычных "алгебраических" приемов блокируются. Допускаются лишь самые простые преобразования стандартизации выражений и утверждений, для которых создаются свои приемы. Такое ограничение алгебраических пребразований объясняется тем, что при выписывании соотношений, связывающих численные параметры рассматриваемых объектов, заранее неизвестно, понадобятся ли вообще эти соотношения. Перечислим приемы, используемые в указанных случаях для работы с синусами:

1. Переход к синусу двойного аргумента.

   $\forall_{ab}(b = 2a \to 2 \sin a \cos a = \sin b)$

   Указатель "эквивалентно" означает, что прием усматривает тождественную истинность равенства в посылках задачи на исследование и заменяет его на константу "истина". Антецедент выделен указателем "идентификатор". Его правая часть обрабатывается нормализатором раскрывания скобок "стандплюс". Уровень срабатывания равен 1.

   $\forall_{ab}(\sin a \cos a = b \to \sin(2a) = 2b)$

Преобразуемый синус двойного аргумента входит в посылку задачи, антецедент является другой посылкой. Выражение a содержит неизвестные, b - не содержит. Уровни срабатывания равны 1,5 и 7.

2. Переход к синусу половинного аргумента.

   $\forall_{ab}(a = b/2 \to \sin b = 2\sin a\cos a)$

   Синус входит в уравнение, являющееся посылкой задачи. Выражение b содержит неизвестные. Антецедент, выделенный указателем "идентификатор", вычисляет половинный угол a, используя нормализатор раскрывания скобок "стандплюс". Проверяется существование уравнения, содержащего a под синусом либо косинусом. Либо таковым оказывается текущее уравнение, либо отсутствует уравнение, имеющее вхождение $\sin b$ или $\cos b$, отличное от преобразуемого вхождения. Указатель "заменавхождений" обеспечивает одновременную замену всех вхождений рассматриваемого синуса в посылки задачи. Уровень срабатывания равен 4.

3. Усмотрение ложного равенства синуса нулю.

   $\forall_a(\neg(\sin a = 0) \to \neg(\sin a = 0))$

   Прием имеет заголовок "второйтерм" и заменяют равенство $\sin a = 0$ на константу "ложь". Это равенство не должно быть расположено под отрицанием. Текущая задача - либо на доказательство, либо на исследование и имеет цель "известно". Антецедент обрабатывается проверочным оператором. Уровень срабатывания равен 1.

4. Сумма единицы и отношения квадратов синуса и косинуса.

   $\forall_a(b(\cos a)^2/(\sin a)^2 + b = b/(\sin a)^2)$

   Уровень срабатывания равен 0.

5. Дробь с суммами синусов в числителе и знаменателе.

   $\forall_{abc}((\sin a + \sin b)/(\sin(c+a) + \sin(c+b)) = \sin(a+b)/(\sin(c+a+b) + \sin c))$

   Выражения a, b содержат символ "угол". Уровень срабатывания равен 5.

6. Выражение неизвестного синуса через косинус.

   $\forall_{ab}(\cos a = b \to \sin a = \sqrt{1-b^2})$

   Антецедент берется в контексте. Выражение a имеет заголовок "угол" либо "уголмежду". Таким образом, оно заключено от 0 до π. Выражение b не содержит символов "угол", "расстояние". Уровень срабатывания равен 8. По этой же теореме создан еще один прием, срабатывающий на уровне 5. Он имеет указатель "теквхожд(1)", определяющий попытку срабатывания при усмотрении посылки вида $\cos a = b$. Заменяемое вхождение синуса ищется в других посылках. Как и выше, a имеет заголовком "угол" либо "уголмежду", b не содержит символов "угол", "расстояние". Дополнительно требуется, чтобы задача имела какие-либо понятия из раздела "физика".

7. Попытка разложения на множители суммы тригонометрических выражений.

   $\forall_{abcdefpqm}(a + b = de \ \& \ (a+b)f = c \ \& \ dp = q \ \& \ m = cp - qef \to m = 0)$

Прием имеет заголовок "вывод". Точка привязки выбрана во втором антецеденте, который идентифицируется с посылкой задачи на доказательство либо на исследование. Каждое содержащее неизвестные слагаемое суммы $a+b$ должно иметь тригонометрическую операцию с неизвестными. Общее число различных тригонометрических аргументов данной суммы должно быть не менее 3. Третий антецедент идентифицируется с некоторым уравнением P, имеющим тригонометрические операции. Выражения c, q должны отличаться от константы 0. Первый антецедент обращается к нормализатору "видумножение" для разложения суммы $a+b$ на множители. Затем находится общий множитель d результата разложения и левой части уравнения P. Проверяется, что он содержит неизвестные. Проверяется также, что результат попытки разложения на множители суммы $a+b$ не имеет заголовка "плюс", имеет более одного тригонометрического аргумента, и каждый такой аргумент уже встречается в некотором уравнении задачи. Левая часть m выводимого уравнения обрабатывается сначала нормализатором раскрывания скобок "стандплюс", а затем - нормализатором ускоренного разложения на множители "факторизация". Проверяется, что полученное выражение содержит неизвестные. Введен средний ограничитель трудоемкости. Уровень срабатывания равен 7.

8. Синус полусуммы углов треугольника равен косинусу половины третьего угла.

   $\forall_{abc}(a+b+c-\pi=0 \to \sin(a/2+b/2)=\cos(c/2))$

   Прием имеет заголовок "второйтерм" и применяется в задачах на преобразование, имеющих цель "учетрезультата" (т.е. при редактировании ответа задачи на описание, имеющей цель "известно"). Уровень срабатывания равен 3.

9. Выражение синуса через известный тангенс половинного аргумента.

   $\forall_{abc}(a-2b=0 \ \& \ \operatorname{tg} b=c \to \sin a=2c/(1+c^2))$

   Прием имеет заголовок "второйтерм". Второй антецедент является посылкой задачи на исследование, имеющей цель "известно". Выражение c не содержит неизвестных. Первый антецедент выделен указателем "идентификатор". Его левая часть обрабатывается нормализаторами общей стандартизации. Уровень срабатывания равен 2.

10. Сокращение на синус половинного аргумента.

    $\forall_{abcde}(d-2b=0 \ \& \ a\sin b=c \to a\sin d=e \leftrightarrow 2c\cos b=e)$

    Преобразуемое уравнение входит в посылку задачи на исследование. Второй антецедент идентифицируется с другой посылкой. Выражение b содержит неизвестные, выражения c, e - не содержат. Первый антецедент, сравнивающий тригонометрические аргументы, выделен указателем "идентификатор". Его левая часть обрабатывается нормализатором раскрывания скобок "стандплюс". Уровень срабатывания равен 2.

11. Вывод следствия с помощью формулы синуса суммы.

    $\forall_{abcde}(a\sin(b+c)+d=e \to a\sin b\cos c+a\sin c\cos b+d=e)$

    Прием имеет заголовок "вывод". Антецедент идентифицируется с посылкой задачи на исследование, имеющей цель "известно". Каждое из выражений $\sin b$, $\sin c$, $\cos b$, $\cos c$ уже встречается в уравнениях задачи. Аргументы b, c содержат неизвестные. Рассматриваемое уравнение не имеет невырожденных (отличных от переменных) числовых атомов. Уровень срабатывания равен 4.

12. Отбрасывание тождеств. Появление тождеств при выводе следствий, например, в планиметрических задачах, наблюдается достаточно часто. Обычно они исключаются в процессе общей стандартизации посылок. Для ускоренного исключения можно создавать специальные приемы. Примером служит следующий прием:

    $\forall_{ab}(\sqrt{2}b\sin(a+\pi/4)=b\sin a+b\cos a$

    Указатель "эквивалентно" определяет не замену левой части равенства на правую, а замену всего равенства на константу "истина". Уровень срабатывания равен 1.

13. Подстановка известного значения синуса.

    $\forall_{ABCa}(\sin(\angle(ABC))=a\rightarrow\sin(\angle(ABC))=a)$

    Прием имеет заголовок "второйтерм". Антецедент идентифицируется с посылкой. Выражение a не содержит неизвестных. Хотя прием и дублирует общелогический прием, выполняющий подстановку значений согласно равенствам из контекста, но бывает полезен для компенсации "слепых пятен", возникающих из-за несовершенства системы переключения внимания. Общелогический прием инициируется равенством - антецедентом, во время рассмотрения которого преобразуемый терм мог еще отсутствовать, а к моменту появления последнего вес "преобразующего" равенства оказывается уже большим. Уровень срабатывания равен 1.

14. Равенство нулю разности синусов.

    $\forall_{ab}(\sin a-\sin b=0\leftrightarrow\sin((a-b)/2)=0\ \vee\ \cos((a+b)/2)=0)$

    Равенство является посылокй задачи на исследование, имеющей цель "известно". Тригонометрический аргумент a содержит числовые атомы, встречающиеся в уравнениях с численными неизвестными. Заменяющая дизъюнкция снабжается комментарием "разборслучаев". Уровень срабатывания равен 5.

15. Равенство синуса нулю.

    $\forall_{ABCa}(\angle(ABC)=a\rightarrow\sin a=0\leftrightarrow a=0)$

    Уровень срабатывания равен 2.

    $\forall_{ab}(a+b=0\rightarrow\sin a=0\leftrightarrow\sin b=0)$

    Антецедент представляет собой посылку задачи на исследование. Выражение a имеет заголовок "плюс", а выражение b - не имеет такого заголовка. Уровень срабатывания равен 2.

16. Переход к косинусу двойного аргумента.

    $\forall_{abx}(a(\sin x)^2=b\cos(2x)\leftrightarrow(2b+a)\cos(2x)=a)$

    Выражение x содержит неизвестные, выражения a,b - не содержат. Уровень срабатывания равен 2.

17. Усмотрение линейного соотношения для синусов.

    $\forall_{abcdpqmn}(ap\sin x/b = mq \;\&\; \neg(q=0) \;\&\; \neg(m=0) \to cp\sin y/d = nq \leftrightarrow adn\sin x = bcm\sin y)$

    Преобразуемое уравнение и первый антецедент идентифицируются с посылками задачи на исследование, имеющей цель "известно". Выражения x, y, а также хотя бы одно из выражений p, q содержат неизвестные. Выражения a, b, c, d, m, n неизвестных не содержат. Выражение q не является произведением синуса неизвестного аргумента на известный коэффициент. Либо p содержит неизвестные, либо q не имеет заголовка "синус", либо $x = y$. Уровень срабатывания равен 4.

18. Разбор случаев для угла, синус которого известен.

    $\forall_{ABCa}(\sin(\angle(ABC)) = a \leftrightarrow \angle(ABC) = \arcsin a \;\vee\; \angle(ABC) = \pi - \arcsin a)$

    Выражение a не содержит неизвестных. Угол $\angle(ABC)$ можно связать по цепочке уравнений с численными неизвестными. Уровень срабатывания равен 10.

19. Определение меньшего из двух смежных углов, для которых известен синус.

    $\forall_{ABCa}(\sin(\angle(ABC)) = a \to \min(\angle(ABC), \pi - \angle(ABC)) = \arcsin a)$

    Идентификация начинается с антецедента, причем выражение a должно не содержать неизвестных. Преобразуемое выражение ищется в других посылках задачи. Уровень срабатывания равен 7.

20. Усмотрение равенства синусов двойного угла.

    $\forall_{xy}(\sin x\cos x = \sin y\cos y \leftrightarrow \sin(x-y) = 0 \;\vee\; \cos(x+y) = 0)$

    Уровень срабатывания равен 4.

Нормализатор общей стандартизации "нормсинус"

В нормализатор вошли следующие приемы, дублирующие приемы общей стандартизации:

1. Синус константы: 0, $pi/2$, $pi/6$, $\pi/4$, $\pi/3$.
2. Синус целого кратного половины пи.
3. Минус под синусом.
4. Аргумент домножен на степень минус единицы.

   $\forall_{an}(n - \text{целое} \to \sin((-1)^n a) = (-1)^n \sin a)$
5. Преобразование дробного слагаемого аргумента к виду суммы дробей.
6. Изменение знака разности под синусом.
7. Формулы приведения.
8. Синус арктангенса.
9. Синус арккосинуса.
10. Синус арксинуса.

11. Синус половины арккосинуса.

12. Учет операции "минугол".

    Выражение "минугол(a)" используется в планиметрии для обозначения меньшего из углов $a, \pi - a$.

    $\forall_{ABC}(\sin \text{минугол}(\angle(ABC)) = \sin\angle(ABC))$

    $\forall_{ABCa}(\text{минугол}(\angle(ABC)) = \text{минугол}(a) \to \sin(\angle(ABC)) = \sin a)$

    Во втором приеме антецедент берется в посылках, причем выражение a не содержит невырожденных числовых атомов.

11.2 Приемы символа "косинус"

Выражение "косинус(a)", прорисовываемое формульным редактором в виде $\cos a$, обозначает синус вещественного числа a. Многие приемы символа "косинус" аналогичны приемам символа "синус". Их мы будем приводить практически без комментариев.

Общая стандартизация выражений

1. Косинус константы.

    (a) $\cos 0 = 1$
    (b) $\cos \pi/2 = 0$
    (c) $\cos \pi/6 = \sqrt{3}/2$
    (d) $\cos \pi/4 = 1/\sqrt{2}$
    (e) $\cos \pi/3 = 1/2$

    Перечисленные приемы имеют уровень срабатывания 0.

    (f) $\cos \pi/5 = (\sqrt{5} + 1)/4$
    (g) $\cos \pi/10 = (\sqrt{10 + 2\sqrt{5}})/4$
    (h) $\cos \pi/8 = \sqrt{2 + \sqrt{2}}/2$
    (i) $\cos \pi/12 = (1 + \sqrt{3})/2\sqrt{2}$

    Если прием применяется при упрощении ответа задачи на вычисление, то уровень срабатывания равен 3, иначе он равен 6.

2. Косинус целого кратного пи.

    $\forall_n(n - \text{целое} \to \cos(\pi n) = (-1)^n)$

    Уровень срабатывания равен 1.

3. Косинус целого кратного половины пи.

    $\forall_n(n - \text{целое} \to \cos(\pi n/2) = ((-1)^{n/2} \text{ при } n - \text{even}, \text{ иначе } 0))$

    Уровень срабатывания равен 1.

4. Минус под косинусом.

   $\forall_a(\cos(-a) = \cos a)$

   Уровень срабатывания равен 0.

5. Модуль под косинусом.

   $\forall_a(\cos|a| = \cos a)$

   Уровень срабатывания равен 1.

6. Аргумент домножен на степень минус единицы.

   $\forall_{an}(n - \text{целое} \rightarrow \cos((-1)^n a) = \cos a)$

   Уровень срабатывания равен 0.

7. Изменение знака разности под косинусом.

   $\forall_{abc}(a = b - c \rightarrow \cos(c - b) = \cos a)$

   Антецедент выделен указателем "идентификатор". Его правая часть обрабатывается нормализаторами общей стандартизации, а также нормализатором "стандупорядочение", выполняющим лексикографическое упорядочение операндов коммутативных операций. Проверяется, что результат лексикографически предшествует выражению $c - b$. Требуется также, чтобы $c - b$ не имело слагаемого вида $m\pi/n$ с натуральными m, n. Это требование продиктовано стандартизацией согласно формалам приведения, в которых устраняются отрицательные рациональные кратные π. Уровень срабатывания равен 1.

8. Преобразование дробного слагаемого аргумента к виду суммы дробей.

   $\forall_{abcdefn}(f = (a + b)^n c/d + e \rightarrow \cos((a + b)^n c/d + e) = \cos f)$

   Аналогично приему для синуса.

9. Формулы приведения.

   $\forall_{abcde}(b = cd + e \;\&\; 0 < c \rightarrow \cos(a + b\pi/d) = (-1)^c \cos(a + e\pi/d))$

   Коэффициенты b, d - натуральные константы. Антецеденты выделены указателем "программа". Прием делает коэффициент при π меньшим единицы. Уровень срабатывания равен 0.

   $\forall_{abcd}(c = 2a - b \;\&\; 0 \leq c \rightarrow \cos(d + a\pi/b) = -\sin(d + c\pi/2b))$

   Коэффициенты a, b - натуральные константы. Прием делает коэффициент при π меньшим одной второй. Уровень срабатывания равен 0.

   $\forall_{abcdefg}(a = bc + e \;\&\; f = b - e \;\&\; g = c + 1 \rightarrow \cos(d - a\pi/b) = (-1)^g \cos(d + f\pi/b))$

   Коэффиценты a, b - натуральные константы. Прием переходит от отрицательного коэффициента при π к положительному. Уровень срабатывания равен 0.

   $\forall_{abcde}(c = 4e - b \;\&\; 0 \leq c \;\&\; d = b - 2e \rightarrow \cos(a + e\pi/b) = \sin(-a + d\pi/2b))$

   Коэффициенты b, e - натуральные. Прием делает коэффициент при π не превосходящим одной четвертой. Уровень срабатывания равен 0.

   $\forall_{an}(n - \text{целое} \rightarrow \cos(n\pi + a) = (-1)^n \cos a)$

   Антецедент обрабатывается проверочным оператором (n может быть неконстантным). Уровень срабатывания равен 0.

10. Переход к половинному аргументу.

    $\forall_{abc}(a\sqrt{1+\cos b}/sqrt2c = a|\cos(b/2)|/c)$

    Уровень срабатывания равен 0.

11. Использование равенства с квадратом синуса.

    $\forall_{ab}(a-(\sin b)^2 = 0 \to a+(\cos b)^2 = 1)$

    Антецедент берется в посылках. Уровень срабатывания равен 1.

12. Сумма произведения тригонометрических функций и функции от суммы либо разности аргументов.

    $\forall_{abc}(c\sin a\sin b + c\cos(a+b) = c\cos a\cos b)$

    $\forall_{abc}(c\cos(a-b) - c\sin a\sin b = c\cos a\cos b)$

    $\forall_{abc}(c\cos(a-b) - c\cos a\cos b = c\sin a\sin b)$

    $\forall_{abc}(c\cos(a+b) - c\cos a\cos b = -c\sin a\sin b)$

    Уровень срабатывания равен 0.

13. Сокращение на косинус половинного аргумента.

    $\forall_{abcdefg}(f = \min(b,c)\ \&\ \neg(1+\cos a = 0) \to e(\sin a)^c/d(1+\cos a)^b + g = e(\sin a)^{c-f}(\sin(a/2))^f/d(1+\cos a)^{b-f}(\cos(a/2))^f + g)$

    $\forall_{abcdefg}(f = \min(b,c)\ \&\ \neg(\sin a = 0) \to d(1+\cos a)^b/e(\sin a)^c + g = d(1+\cos a)^{b-f}(\cos(a/2))^f/e(\sin a)^{c-f}(\sin(a/2))^f + g)$

    Дробь входит в условие задачи на преобразование. Показатели степени b, c - натуральные константы. Либо $b = c$ и условие задачи не имеет других вхождений тригонометрического аргумента a, либо существует вхождение тригонометрического аргумента, четным кратным которого является a. Первый антецедент выделен указателем "программа", второй - обрабатывается проверочным оператором. Первый прием выводит для сопровождения по о.д.з. утверждение $\neg(\cos(a/2) = 0)$, второй - аналогичное утверждение с синусом. При преобразовании подынтегрального выражения приемы блокируются. Уровень срабатывания равен 2.

14. Преобразование подынтегрального выражения. Так как приемы этого подраздела аналогичны рассмотренным ранее приемам для синуса, ограничиваемся указанием их теорем.

    (a) Понижение степени числителя путем выражения квадрата косинуса через квадрат синуса.

    $\forall_{abcde}(a(\cos e)^c/b(\sin e)^d = a(\cos e)^{c-2}/b(\sin e)^d - a(\cos e)^{c-2}/b(\sin e)^{d-2})$

    (b) Преобразование к виду суммы тригонометрических выражений.

    $\forall_{abcdep}(p = a(\cos b)^c/d + e \to a(\cos b)^c/d + e = p)$

    $\forall_{abcdep}(p = a\cos b\cos c/d + e \to a\cos b\cos c/d + e = p)$

    (c) Произведение тригонометрических функций, разность либо сумма аргументов которых не зависит от переменной интегрирования.

    $\forall_{abcdefg}(d/e(\cos(c+a))^f(\cos(c+b))^g = d\sin(c+a)/e\sin(a-b)(\cos(c+a))^f(\cos(c+b))^{g-1} - d\sin(c+b)/e\sin(a-b)(\cos(c+a))^{f-1}(\cos(c+b))^g)$

   $\forall_{abcdefg}(d/e(\cos(a-c))^f(\cos(c+b))^g = d\sin(a-c)/e\sin(a+b)(\cos(a-c))^f(\cos(c+b))^{g-1} + d\sin(c+b)/e\sin(a+b)(\cos(a-c))^{f-1}(\cos(c+b))^g)$

   (d) Устранение двойного аргумента.
   $\forall_{ab}(a = b/2 \to \cos b = 2(\cos a)^2 - 1)$

   (e) Переход к одной трети тригонометрического аргумента.
   $\forall_{ab}(a = 3b \to \cos a = 4(\cos b)^3 - 3\cos b)$

15. Редактирование ответа задачи на вычисление.

   (a) Сложение синуса и косинуса одинаковых аргументов.
   $\forall_{ab}(a\sin b + a\cos b = \sqrt{2}a\sin(b + \pi/4))$

   Выражение входит в условие задачи на преобразование, имеющей цель "известны". Уровень срабатывания равен 3.

   (b) Переход к тангенсам в тригонометрических дробях специального вида.
   $\forall_{abcden}(d(\cos b)^n/e(a\cos b + c\sin b) = d(\cos b)^{n-1}/e(a + c\,\mathrm{tg}\, b))$

   Выражение входит в условие задачи на преобразование, имеющей цель "учетрезультата". Допускается перестановка числителя и знаменателя. Уровень срабатывания равен 3.

   (c) Переход к котангенсам в тригонометрических дробях специального вида.
   $\forall_{abcden}(d(\sin b)^n/e(a\sin b + c\cos b) = d(\sin b)^{n-1}/e(a + c\,\mathrm{ctg}\, b))$
   $\forall_{abcdef}(d\sqrt{(b(\sin a)^2 + c(\cos a)^2)f}/e\sin a = d\sqrt{(b + c(\mathrm{ctg}\, a)^2)f}/e)$
   Аналогично предыдущему пункту.

   (d) Произведение косинусов углов со слагаемым одна шестая пи.
   $\forall_a(\cos(a + \pi/6)\cos(\pi/6 - a) = (1 + 2\cos(2a))/4)$

   Произведение входит в условие задачи на преобразование, имеющей цель "учетрезультата". Уровень срабатывания равен 3.

   (e) Приведение подобных членов с косинусами.
   $\forall_{abc}(a\cos b + c\cos b = (a + c)\cos b)$

   Сумма входит в условие задачи на преобразование, имеющей цель "учетрезульата". Уровень срабатывания равен 1.

Общая стандартизация утверждений

1. Равенство нулю косинуса угла, заключенного между 0 и π.

   $\forall_a(0 \le a \;\&\; 0 \le \pi - a \to \cos a = 0 \leftrightarrow a = \pi/2)$

   Антецеденты обрабатываются проверочными операторами. Попытка проверки предпринимается лишь в особых случаях - для условий задач на описание, имеющих цель "неравенства". Такие задачи используются при определении области интегрирования.

2. Преобразование неравенства для косинуса в равенство.

   $\forall_a(0 \le \cos a - 1 \leftrightarrow \cos a = 1)$

   Уровень срабатывания равен 0.

3. Усмотрение равенства косинуса выражению, модуль которого больше единицы.

   $\forall_{abc}(c \le |b| \;\&\; 0 < c - 1 \to \neg(\cos a = b))$

   Равенство входит в условие задачи на описание. Выражение a содержит неизвестные, b - не константное и не содержит неизвестных. Первый антецедент выделен указателем "значения". Он обращается к синтезатору "нижняяоценка" и определяет константную нижнюю оценку c модуля b. Второй антецедент, обрабатываемый проверочным оператором, убеждается, что эта оценка больше единицы. Введен достаточно сильный ограничитель трудоемкости. Уровень срабатывания равен 1.

Косинус выражения, содержащего обратные тригонометрические функции

1. Косинус арксинуса.

   $\forall_a(0 \le 1 + a \;\&\; 0 \le 1 - a \to \cos \arcsin a = \sqrt{1 - a^2})$

   Антецеденты обрабатываются проверочными операторами. Для сопровождения по о.д.з. выводится утверждение $0 \le 1 - a^2$. Если выражение a константное, то уровень срабатывания равен 0, иначе - равен 1. Если сопровождение по о.д.з. может нарушаться, используются два других приема, созданные по одной и той же теореме $\forall_a(\cos \arcsin a = \sqrt{1 - a^2})$. Первый из них применяется в задачах на преобразование, имеющих цель "известны" (редактирование ответа задачи на вычисление), второй - в задачах на исследование, имеющих цель "известно". Уровни срабатывания равны, соответственно, 2 и 0.

2. Косинус арккосинуса.

   $\forall_a(\cos \arccos a = a)$

   Прием применяется даже к утверждениям, обеспечивающим сопровождение по о.д.з. Уровень срабатывания равен 0.

3. Косинус арктангенса.

   $\forall_a(\cos \operatorname{arctg} a = 1/\sqrt{a^2 + 1})$

   Как и выше, преобразуются даже утверждения сопровождения по о.д.з. В задаче на исследование, имеющей цель "известно", прием срабатывает на уровне 0, иначе - на уровне 1.

4. Косинус половины арккосинуса.

   $\forall_a(0 \le 1 + a \;\&\; 0 \le 1 - a \to \cos(\arccos a/2) = \sqrt{(1 + a)/2})$

   Антецеденты обрабатываются проверочными операторами. Уровень срабатывания равен 1.

   $\forall_a(\cos(\arccos a/2 + \pi/4) = (\sqrt{1 + a} - \sqrt{1 - a})/2)$

   Прием применяется в задаче на преобразование, имеющей цель "учетрезультата" либо цель "известны". Уровень срабатывания равен 1.

5. Косинус половины арксинуса.

   $\forall_a(0 \le 1 + a \;\&\; 0 \le 1 - a \to \cos(\arcsin a/2) = \sqrt{(1 + \sqrt{1 - a^2})/2})$

   Уровень срабатывания равен 1.

6. Косинус половины арктангенса.

   $\forall_a(\cos(\operatorname{arctg} a/2) = \sqrt{(\sqrt{1+a^2}+1)/2\sqrt{1+a^2}})$

   Выражение находится в условии задачи. Уровень срабатывания равен 2.

7. Косинус одной четвертой арккосинуса.

   $\forall_a(\cos(\arccos a/4) = \sqrt{(1+\sqrt{(1+a)/2})/2})$

   Выражение находится в посылке задачи на исследование, имеющей цель "известно". Выражение a константное. Уровень срабатывания равен 0.

   $\forall_a(\cos(\pi/4+\arccos a/4) = (\sqrt{1+\sqrt{(1+a)/2}} - \sqrt{1-\sqrt{(1+a)/2}})/2)$

   Аналогично предыдущему.

8. Косинус трех вторых арккосинуса.

   $\forall_a(\cos(3\arccos a/2) = (2a-1)\sqrt{(a+1)/2})$

   Выражение находится в условии задачи. Уровень срабатывания равен 2.

9. Косинус удвоенного арксинуса.

   $\forall_a(\cos(2\arcsin a) = 1-2a^2)$

   Уровень срабатывания равен 0.

10. Косинус удвоенного арккосинуса.

    $\forall_a(\cos(2\arccos a) = 2a^2-1)$

    Аналогично предыдущему.

11. Косинус удвоенного арктангенса.

    $\forall_a(\cos(2\operatorname{arctg} a) = (1-a^2)/(1+a^2))$

    Уровень срабатывания равен 1.

12. Косинус тройного арксинуса.

    $\forall_a(0 \le a+1 \ \& \ 0 \le 1-a \to \cos(3\arcsin a) = (1-4a^2)\sqrt{1-a^2})$

    Для сопровождения по о.д.з. выводится утверждение $0 \le 1-a^2$. Если выражение a константное, то уровень срабатывания равен 0, иначе он равен 1.

13. Косинус тройного арккосинуса.

    $\forall_a(\cos(3\arccos a) = 4a^3-3a)$

    Уровень срабатывания равен 0.

14. Косинус учетверенного арктангенса.

    $\forall_a(\cos(4\operatorname{arctg} a) = (a^4-6a^2+1)/(1+a^2)^2)$

    Выражение входит в условие задачи. Уровень срабатывания равен 1.

15. Косинус от линейной комбинации с участием обратной тригонометрической функции.

    $\forall_{ab}(\cos(a+b) = \cos a \cos b - \sin a \sin b)$

    Аналогично случаю синуса.

    $\forall_{ab}(\cos(\arccos a + b) = a \cos b - \sqrt{1-a^2} \sin b)$

    Выражение входит в посылку задачи на исследование, имеющей цель "известно". Выражения a, b не содержат неизвестных.

16. Обращение к вспомогательной задаче для преобразования косинуса от выражения, содержащего обратные тригонометрические функции.

    $\forall_{ab}(a = \cos b \to \cos b = a)$

    Аналогично случаю синуса.

Уравнения

1. Простейшее уравнение для косинуса.

   $\forall_{ab}(\cos a = b \leftrightarrow 0 \leq b+1 \ \&\ 0 \leq 1-b \ \&\ (\exists_n(n - \text{целое} \ \&\ a = \arccos b + 2\pi n) \vee \exists_n(n - \text{целое} \ \&\ a = -\arccos b + 2\pi n)))$

   Уравнение является условием задачи на описание. Уровень срабатывания приема равен 2.

   $\forall_a(\cos a = 0 \leftrightarrow \exists_n(n - \text{целое} \ \&\ a = \pi(2n+1)/2))$

   $\forall_a(\cos a = 1 \leftrightarrow \exists_n(n - \text{целое} \ \&\ a = 2\pi n))$

   $\forall_{ab}(\cos a = \cos b \leftrightarrow \exists_n(a = b + 2\pi n \ \&\ n - \text{целое}) \vee \exists_n(a = -b + 2\pi n \ \&\ n - \text{целое}))$

   $\forall_{ab}(\cos a = \sin b \leftrightarrow \exists_n(a = \pi/2 - b + 2\pi n \ \&\ n - \text{целое}) \vee \exists_n(a = b - \pi/2 + 2\pi n \ \&\ n - \text{целое}))$

   $\forall_{ab}(\cos a = -\sin b \leftrightarrow \exists_n(a = \pi/2 + b + 2\pi n \ \&\ n - \text{целое}) \vee \exists_n(a = -b - \pi/2 + 2\pi n \ \&\ n - \text{целое}))$

   $\forall_{ab}(\cos a = -\cos b \leftrightarrow \exists_n(a = b + \pi + 2\pi n \ \&\ n - \text{целое}) \vee \exists_n(a = -b + \pi + 2\pi n \ \&\ n - \text{целое}))$

   Уравнение - условие задачи на описание. Уровень срабатывания равен 1. Первый прием продублирован: та его версия, которая использует оператор фильтра "фильтрсерии", срабатывает на уровне 1, другая - на уровне 6.

   $\forall_{ab}(0 \leq a \ \&\ 0 \leq \pi - a \to \cos a = b \leftrightarrow 0 \leq b+1 \ \&\ 0 \leq 1-b \ \&\ a = \arccos b)$

   $\forall_{ab}(0 \leq a - \pi \ \&\ 0 \leq 2\pi - a \to \cos a = b \leftrightarrow 0 \leq b+1 \ \&\ 0 \leq 1-b \ \&\ a = 2\pi - \arccos b)$

   Уравнение - условие задачи на описание, имеющей цель "стандравно" либо "областьграницы". Уровень срабатывания равен 0.

   $\forall_{ab}(0 \leq \pi + a \ \&\ 0 \leq \pi - a \to \cos a = b \leftrightarrow 0 \leq b+1 \ \&\ 0 \leq 1-b \ \&\ (a = \arccos b \vee a = -\arccos b))$

   Уравнение - условие задачи на описание, имеющей цель "стандравно". Уровень срабатывания равен 1.

2. Умножение на синус произведения косинусов кратных аргументов. Этот хорошо известный прием позволяет упрощать уравнения с произведениями вида $\cos a \cos 2a \cos 4a \ldots$ путем домножений обеих частей на $\sin a$ и многократного применения формулы синуса двойного аргумента.

   $\forall_{abcde}(e = 2a \to b\cos a \cos e + d = c \leftrightarrow b\sin(4a) + 4d\sin a = 4c\sin a \;\&\; \neg(\sin a = 0) \;\vee\; \sin a = 0 \;\&\; b\cos a\cos e + d = c)$

   Уравнение является условием задачи на описание. Выражение a содержит неизвестные, c - не содержит. Либо выражение d - вырожденное, равное нулю, либо оно состоит из единственного слагаемого, а c равно 0. Антецедент сравнивает аргументы a, e. Он выделен указателем "идентификатор", причем правая часть обрабатывается нормализатором "стандплюс". Проверяется, что b делится на $\cos 4a$, но не делится на $\cos(a/2)$. Должно отсутствовать другое уравнение задачи, имеющее общую неизвестную с выражением a. Заменяющее утверждение снабжается комментарием "разборслучаев". Уровень срабатывания равен 0.

3. Попытка преобразования линейной комбинации известных синуса и косинуса в правой части уравнения к виду синуса либо косинуса.

   $\forall_{abcdx}(d = ab + ac \to \cos x = ab + ac \leftrightarrow \cos x = d)$

   Уравнение является условием задачи на описание. Выражение x содержит неизвестные, правая часть уравнения неизвестных не содержит. Каждое из выражений b, c имеет вид $\pm\sin(\ldots)$ либо $\pm\cos(\ldots)$. Антецедент выделен указателем "идентификатор", его правая часть обрабатывается нормализатором "видумножение". Проверяется, что d имеет вид $\pm\sin(\ldots)$ либо $\pm\cos(\ldots)$. Уровень срабатывания равен 1.

4. Применение формулы косинуса суммы для перехода к уже имеющемуся неизвестному тригонометрическому аргументу.

   $\forall_{ab}(\cos(a + b) = \cos a \cos b - \sin a \sin b)$

   Косинус входит в условие задачи на описание. Переменная b идентифицируется с суммой всех содержащих неизвестные слагаемых аргумента. Эта сумма и остаточная сумма a невырожденные. Условие уже содержит тригонометрическюу операцию от аргумента b. Уровень срабатывания равен 6.

5. Линейная комбинация уравнений с целью исключения неизвестного косинуса. Прием применяется в задачах по планиметрии для получения соотношения, связывающего численные неизвестные.

   $\forall_{abcdefg}(a + b\cos c = d \;\&\; e + f\cos c = g \to (a - d)f = (e - g)b)$

   Заголовок приема - "вывод". Антецеденты идентифицируются с посылками задачи на доказательство либо задачи на исследование, имеющей цель "известно". Выражение c содержит неизвестные. Ни одно из выражений a, b, d, e, f, g не имеет тригонометрических операций. Выводимое утверждение содержит неизвестные и не содержит символов "расстояние", "угол". В задачах по физике прием блокируется. Уровень срабатывания равен 10.

6. Применение формулы произведения косинусов для перехода в уравнении к общему неизвестному тригонометрическому аргументу.

   $\forall_{abcd}(c = \cos(a - b) \;\&\; d = \cos(a + b) \to \cos a \cos b = (c + d)/2)$

Произведение входит в условие задачи на описание. Аргументы a, b содержат неизвестные. Антецеденты выделены указателем "идентификатор". Их правые части обрабатываются нормализаторами общей стандартизации. Хотя бы один из результатов c, d не содержит неизвестных. Каждый расположенный в уравнении вне рассматриваемого произведения неизвестный тригонометрический аргумент равен, с точностью до знака, одному из аргументов c, d. Уровень срабатывания равен 6.

7. Переход к уравнению относительно синуса либо косинуса неизвестного выражения.

   $\forall_{abcdeAP}((d(\cos a)^b + c = e) = P(\cos a)\ \&\ (P(y)\ \&\ y - \text{число}\ \&\ |y| \leq 1) = A(y) \rightarrow d(\cos a)^b + c = e \leftrightarrow A(\cos a))$

   $\forall_{abcdeAP}((d(\sin a)^b + c = e) = P(\sin a)\ \&\ (P(y)\ \&\ y - \text{число}\ \&\ |y| \leq 1) = A(y) \rightarrow d(\sin a)^b + c = e \leftrightarrow A(\sin a))$

   Уравнение является условием задачи на описание, содержащим единственную неизвестную. Выражение a содержит эту неизвестную. Антецеденты выделены указателем "идентификатор". Первый из них усматривает возможность представить уравнение в таком виде, что все вхождения неизвестных заключены только внутри вхождений выражения $\cos a$. Указатель "новаргумент(x40 x23 извлечение)" уточняет, что при этом используется нормализатор "извлечение". Во втором антецеденте рассматривается результат $P(y)$ переобозначения в текущем уравнении выражения $\cos a$ на новую неизвестную y. С помощью вспомогательной задачи он разрешается относительно y, дополнительно учитывается условие $|y| \leq 1$. Заменяющее утверждение получается из ответа данной задачи подстановкой вместо y выражения $\cos a$. Уровень срабатывания равен 5.

   $\forall_{abcxy}(y - 2x = 0 \rightarrow a\cos x + b\cos y = c \leftrightarrow a\cos x + 2b(\cos x)^2 = b + c)$

   $\forall_{abcxy}(y - 2x = 0 \rightarrow a\sin x + b\cos y = c \leftrightarrow a\sin x - 2b(\sin x)^2 = c - b)$

   Уравнение является условием задачи на описание. Выражение x содержит неизвестные, выражения a, b, c - не содержат. Антецедент выделен указателем "идентификатор". Он проверяет, что аргумент y равен $2x$, используя для этого нормализатор "стандплюс". Уровень срабатывания равен 2.

8. Подбор примера.

   $\forall_x(x = 0 \rightarrow \neg(\cos x = 0))$

   Прием имеет заголовок "подборзначений". Отрицание равенства является условием задачи на описание, имеющей цель "пример". Переменная x идентифицируется с неизвестной. Антецедент выделен указателем "подборзначений". Он замещает отрицание равенства во вспомогательной задаче. Уровень срабатывания равен 3.

9. Существование решения уравнения.

   $\forall_a(a - \text{число} \rightarrow \exists_x(x - \text{число}\ \&\ a = \cos x) \leftrightarrow -1 \leq a\ \&\ a \leq 1)$

   Прием имеет заголовок "связка". Если задача на описание имеет только два условия с неизвестной x - "$x - \text{число}$" и "$a = \cos x$", причем эта неизвестная - несущественная, то вместо данных условий помещается условие $-1 \leq a\ \&\ a \leq 1$. Уровень срабатывания равен 0.

10. Отрицания равенств с неизвестным косинусом.

    (a) Явное разрешение относительно неизвестной.

    $\forall_{abcdef}(bd = 2f \,\&\, e -$ целое $\rightarrow \neg(\cos(a\pi/b + ce\pi/d) = 0) \leftrightarrow \exists_g(g \in \{0, \ldots, d-1\} \,\&\, \neg(2f|ad + cbg - f) \,\&\, \exists_n(n -$ целое $\&\, e = dn + g)))$

    $\forall_{abcd}(b = 2d \,\&\, c -$ целое $\rightarrow \neg(\cos(ac\pi/b) = 0) \leftrightarrow \exists_e(e \in \{0, \ldots, b-1\} \,\&\, \neg(b|ae - d) \,\&\, \exists_n(n -$ целое $\&\, c = bn + e)))$

    Аналогично случаю синуса.

    (b) Группировка отрицаний равенства нулю косинусов суммы и разности.

    $\forall_{ab}(\neg(\cos(a + b) = 0) \,\&\, \neg(\cos(a - b) = 0) \leftrightarrow \neg(\cos(2a) + \cos(2b) = 0))$

    Прием имеет заголовок "заменаусловия(второйтерм)". Отрицания суть условия задачи на описание, не используемые для сопровождения по о.д.з. Хотя бы одно из выражений a, b содержит неизвестные. Уровень срабатывания равен 1.

    (c) Группировка отрицаний равенства нулю суммы и разности косинуса и единицы.

    $\forall_a(\neg(1 - \cos a = 0) \,\&\, \neg(1 + \cos a = 0) \leftrightarrow \neg(\sin a = 0))$

    Аналогично предыдущему.

    (d) Исключение избыточного отрицания равенства косинуса нулю.

    $\forall_{ab}(a = 2b \,\&\, \neg(\sin a = 0) \rightarrow \neg(\cos b = 0))$

    Прием имеет заголовок "второйтерм". Равенство косинуса нулю входит в условие задачи на описание и не используется для сопровождения по о.д.з. Выражение b содержит неизвестные. Второй антецедент идентифицируется с условием, первый - выделен указателем "идентификатор". Уровень срабатывания равен 1.

    $\forall_{abc}(a = 2b \,\&\, \sin a < c \,\&\, c \leq 0 \rightarrow \neg(\cos b = 0))$

    $\forall_{abc}(a = 2b \,\&\, c < \sin a \,\&\, 0 \leq c \rightarrow \neg(\cos b = 0))$

    Аналогично предыдущему. Последний антецедент обрабатывается проверочным оператором.

11. Переход к двойному аргументу в уравнении, определяющем значение квадрата косинуса.

    $\forall_{ax}((\cos x)^2 = a \leftrightarrow \cos(2x) = 2a - 1)$

    Уравнение является условием задачи на описание. Выражение x содержит неизвестные, выражение a - не содержит и отлично от константы ноль. Отсутствует условие задачи, имеющее вид $0 \leq x$. Уровень срабатывания равен 0.

12. Применение формул синуса либо косинуса суммы для перехода к уравнению, левая часть которого - линейная комбинация неизвестных синуса и косинуса.

    $\forall_{abcdx}(a\sin(b + x) + c = d \leftrightarrow a\sin x\cos b + a\cos x\sin b + c = d)$

    $\forall_{abcdx}(a\cos(b + x) + c = d \leftrightarrow a\cos x\cos b - a\sin x\sin b + c = d)$

    Уравнение является условием задачи на описание. Выражение x содержит неизвестные, выражения a, b, d - не содержат. Каждое неизвестное слагаемое выражения c имеет вид $k\sin(x+m)$ либо $k\cos(x+m)$ при некоторых известных k, m. Уровень срабатывания равен 4.

13. Декомпозиция уравнения, выполнимого только для экстремальных значений синуса или косинуса.

    $\forall_{abcdxy}(|a| < |b| \ \& \ |a| < |c| \ \& \ -d+a\mathrm{sg}(b)\mathrm{sg}(c)+(|b|+|c|)\mathrm{sg}(d) = 0 \rightarrow a(\sin x)^n(\sin y)^m + b(\sin x)^n + c(\sin y)^m = d \leftrightarrow (\sin x)^n = \mathrm{sg}(b)\mathrm{sg}(d) \ \& \ (\sin y)^m = \mathrm{sg}(c)\mathrm{sg}(d))$

    $\forall_{abcdxy}(|a| < |b| \ \& \ |a| < |c| \ \& \ -d+a\mathrm{sg}(b)\mathrm{sg}(c)+(|b|+|c|)\mathrm{sg}(d) = 0 \rightarrow a(\cos x)^n(\sin y)^m + b(\cos x)^n + c(\sin y)^m = d \leftrightarrow (\cos x)^n = \mathrm{sg}(b)\mathrm{sg}(d) \ \& \ (\sin y)^m = \mathrm{sg}(c)\mathrm{sg}(d))$

    Уравнение является условием задачи на описание. Переменные a, b, c, d идентифицируются с десятичными константами, переменные m, n - с натуральными константами. Выражения x, y содержат неизвестные. Указатель "альтернатива" позволяет рассматривать также случай, когда y находится под косинусом. Уровень срабатывания равен 5.

    $\forall_{abn}(n - \mathrm{rational} \ \& \ \text{числитель}(n) - \mathrm{even} \ \& \ b \leq 0 \ \& \ 0 \leq a \rightarrow a(\cos x)^n + b = a \leftrightarrow (\cos x)^2 = 1 \ \& \ b = 0)$

    Уравнение является условием задачи на описание. Выражение x содержит неизвестные, выражение a - не содержит. Переменная n идентифицируется с константой. Указатель "альтернатива" позволяет рассматривать также случай синуса. Уровень срабатывания равен 1.

14. Применение формулы косинуса разности для перехода к общему неизвестному аргументу.

    $\forall_{axy}(x + y = a \rightarrow \cos x = \cos y \cos a + \sin y \sin a)$

    Косинус входит в условие задачи на описание. Выражение x содержит неизвестные и имеет заголовок "плюс". Указатель "контекст(условие(x2)тригаргумент(x2 x24))" определяет идентификацию в некотором условии задачи тригонометрического аргумента y, отличного от x. Он должен содержать неизвестные. Проверяется, что текущее уравнение не имеет неизвестных тригонометрических аргументов, отличных от x, y. Антецедент выделен указателем "идентификатор". Его левая часть обрабатывается нормализатором "нормплюс". Результат a не должен содержать неизвестных. Уровень срабатывания равен 4.

15. Перенесение во внешнюю задачу на описание равенства, определяющего значение неизвестного косинуса. Если при выводе следствий в блоке анализа задачи на описание появляется равенство неизвестного косинуса известному выражению, то данное равенство переносится в задачу на описание и реализуется возвращение к ее рассмотрению. Теорема приема имеет вид "замещениеусловий(равно(косинус(x1)x2))". Заголовок приема - "замещениеусловий". Прием инициируется идентификацией равенства $\cos a = b$. Это равенство должно быть посылкой задачи на исследование, не имеющей цели "известно". Выражение a содержит неизвестные, b - не содержит. Если a содержит единственную неизвестную x, то внешняя задача на описание не должна иметь условия, представляющего собой равенство тригонометрической операции, содержащей x, известному выражению. Утверждение $\cos a = b$ присоединяется к условиям внешней задачи на описание. Указатель "обрывзадачи" определяет обрыв сканирования текущей задачи на исследование и возвращение к сканированию внешней задачи.

Неравенства

1. Решение простейших нестрогих неравенств с косинусом.

   $\forall_{ab}(\cos b \leq a \leftrightarrow 0 \leq a{-}1 \vee a{-}1 \leq 0 \,\&\, 0 \leq 1{+}a \,\&\, \exists_n(n{-}\text{целое} \,\&\, 2\pi n{+}\arccos a \leq b \,\&\, b \leq 2\pi n + 2\pi - \arccos a))$

   $\forall_{ab}(a \leq \cos b \leftrightarrow 1{+}a \leq 0 \vee 0 \leq 1{-}a \,\&\, 0 \leq 1{+}a \,\&\, \exists_n(n{-}\text{целое} \,\&\, 2\pi n{-}\arccos a \leq b \,\&\, b \leq 2\pi n + \arccos a))$

   Приемы аналогичны случаю синуса. Используется оператор "фильтрпромежутков", уровни срабатывания равны 2 и 6.

   $\forall_{ab}(0 \leq b \,\&\, 0 \leq \pi{-}b \rightarrow \cos b \leq a \leftrightarrow 0 < a{-}1 \vee 0 \leq 1{+}a \,\&\, 0 \leq 1{-}a \,\&\, \arccos a \leq b)$

   $\forall_{ab}(0 \leq b \,\&\, 0 \leq \pi{-}b \rightarrow a \leq \cos b \leftrightarrow a{+}1 < 0 \vee 0 \leq 1{+}a \,\&\, 0 \leq 1{-}a \,\&\, b \leq \arccos a)$

   Неравенство является условием задачи на описание, не используемым для сопровождения по о.д.з. Задача решается для определения области интегрирования и не имеет уравнений. Выражение b содержит неизвестные, a - не содержит. Уровень срабатывания равен 1. Если задача не имеет других условий, представляющих собой неравенства с тригонометрическим аргументом b, то добавляются еще два приема, имеющих тот же уровень срабатывания:

   $\forall_{ab}(0 \leq \pi + b \,\&\, 0 \leq \pi - b \rightarrow a \leq \cos b \leftrightarrow a + 1 < 0 \;\vee\; 0 \leq 1 + a \,\&\, 0 \leq 1 - a \,\&\, -\arccos a \leq b \,\&\, b \leq \arccos a)$

   $\forall_{ab}(0 \leq \pi + b \,\&\, 0 \leq \pi - b \rightarrow \cos b \leq a \leftrightarrow 0 < a - 1 \;\vee\; 0 \leq 1 + a \,\&\, 0 \leq 1 - a \,\&\, (\arccos a \leq b \;\vee\; b \leq -\arccos a))$

2. Решение простейших строгих неравенств с косинусом. Аналогично предыдущему:

   $\forall_{ab}(\cos b < a \leftrightarrow 0 < a{-}1 \vee a{-}1 \leq 0 \,\&\, 0 \leq 1{+}a \,\&\, \exists_n(n{-}\text{целое} \,\&\, 2\pi n{+}\arccos a < b \,\&\, b < 2\pi n + 2\pi - \arccos a))$

   $\forall_{ab}(a < \cos b \leftrightarrow 1{+}a < 0 \vee 0 \leq 1{-}a \,\&\, 0 \leq 1{+}a \,\&\, \exists_n(n{-}\text{целое} \,\&\, 2\pi n{-}\arccos a < b \,\&\, b < 2\pi n + \arccos a))$

   Уровни срабатывания равны 2 и 6.

   $\forall_{ab}(0 \leq b \,\&\, 0 \leq \pi{-}b \rightarrow \cos b < a \leftrightarrow 0 < a{-}1 \vee 0 \leq 1{+}a \,\&\, 0 \leq 1{-}a \,\&\, \arccos a < b)$

   $\forall_{ab}(0 \leq b \,\&\, 0 \leq \pi{-}b \rightarrow a < \cos b \leftrightarrow a{+}1 < 0 \vee 0 \leq 1{+}a \,\&\, 0 \leq 1{-}a \,\&\, b < \arccos a)$

   Уровень срабатывания равен 1.

3. Решение двусторонних неравенств с косинусом.

   $\forall_{abc}(0 \leq 1 - a \,\&\, 0 \leq 1 + a \,\&\, 0 \leq 1 - b \,\&\, 0 \leq 1 + b \rightarrow a \leq \cos c \,\&\, \cos c \leq b \leftrightarrow 0 < b - a \,\&\, (\exists_n(n - \text{целое} \,\&\, 2\pi n + \arccos b \leq c \,\&\, c \leq 2\pi n + \arccos a) \;\vee\; \exists_n(n - \text{целое} \,\&\, 2\pi n - \arccos a \leq c \,\&\, c \leq 2\pi n - \arccos b)))$

   $\forall_{abc}(0 \leq 1 - a \,\&\, 0 \leq 1 + a \,\&\, 0 \leq 1 - b \,\&\, 0 \leq 1 + b \rightarrow a \leq \cos c \,\&\, \cos c \leq b \leftrightarrow 0 \leq b - a \,\&\, (\exists_n(n - \text{целое} \,\&\, 2\pi n + \arccos b \leq c \,\&\, c \leq 2\pi n + \arccos a) \;\vee\; \exists_n(n - \text{целое} \,\&\, 2\pi n - \arccos a \leq c \,\&\, c \leq 2\pi n - \arccos b)))$

   Приемы имеет заголовок "заменаусловия(второйтерм)". Неравенства идентифицируются с условиями задачи на описание, не имеющей уравнений. Первый прием снабжен указателями "альтернатива(...)", разрешающими вместо нестрогих неравенств рассматривать строгие. При этом требуется, чтобы хотя бы одно из неравенств было строгим. Второй прием ориентирован на случай, когда оба неравенства нестрогие. В остальном приемы аналогичны случаю синуса. Уровни срабатывания равны 1 и 5.

$\forall_{abc}(0 \leq 1-a \,\&\, 0 \leq 1+a \,\&\, 0 \leq 1-b \,\&\, 0 \leq 1+b \,\&\, 0 \leq \pi+c \,\&\, 0 \leq \pi-c \rightarrow a \leq \cos c \,\&\, \cos c \leq b \leftrightarrow 0 \leq b-a \,\&\, (\arccos b \leq c \,\&\, c \leq \arccos a \ \vee\ -\arccos a \leq c \,\&\, c \leq -\arccos b))$

Неравенства суть условия задачи на описание, не имеющей уравнений. Они не используются для сопровождения по о.д.з. Выражение c содержит неизвестные, выражения a, b - не содержат. Оператор "фильтрпромежутков" ложен. Другие неравенства для $\cos c$ отсутствуют. Уровень срабатывания равен 1.

4. Переход от двустороннего неравенства для косинуса и отрицания равенства к дизъюнкции двух двусторонних неравенств.

   $\forall_{abcde}(a+d/e < 0 \,\&\, 0 < b+d/e \,\&\, \neg(e=0) \rightarrow a \leq \cos c \,\&\, \cos c \leq b \,\&\, \neg(e\cos c+d = 0) \leftrightarrow a \leq \cos c \,\&\, \cos c < -d/e \ \vee\ -d/e < \cos c \,\&\, \cos c \leq b)$

   Неравенства и отрицание равенства суть условия задачи на описание, не имеющей уравнений. Выражение c содержит неизвестные, выражения a, b, d, e - не содержат. Задача не имеет других условий с неизвестными тригонометрическими аргументами. Указатели "альтернатива(...)" разрешают рассмотрение строгих неравенств. Выводимая дизъюнкция снабжается комментарием "разборслучаев". Уровень срабатывания равен 0.

5. Преобразование отрицания равенства с неизвестным косинусом в дизъюнкцию двух неравенств.

   $\forall_{ax}(\neg(\cos x = a) \leftrightarrow \cos x < a \ \vee\ a < \cos x)$

   Отрицание является условием задачи на описание, не имеющей уравнений. Выражение x содержит неизвестные, выражение a - не содержит. Задача имеет неравенство, в одной части которого содержится либо неизвестная, либо тригонометрическая функция неизвестного аргумента, а другая часть известна. Выводимая дизъюнкция снабжается комментарием "разборслучаев". Уровень срабатывания равен 1.

   $\forall_{abx}(\neg(a=0) \rightarrow \neg(a\cos x + b = 0) \leftrightarrow \cos x < -b/a \ \vee\ -b/a < \cos x)$

   Аналогично предыдущему.

6. Неравенство для произведения косинусов разности и суммы.

   $\forall_{ab}(0 < \cos(a+b)\cos(a-b) \leftrightarrow 0 < \cos(2a) + \cos(2b))$

   Неравенство является условием задачи на описание, не имеющей уравнений. Оно не используется для сопровождения по о.д.з. Одно из выражений a, b содержит неизвестные, другое - не содержит. Указатели приема разрешают рассматривать нестрогое неравенство и переставлять части неравенства. Уровень срабатывания равен 1.

7. Существование решения неравенства с линейной комбинацией неизвестных синуса и косинуса.

   $\forall_{abcpqr}(\neg(p=0) \rightarrow \exists_x(x - \text{число} \,\&\, a\sin(px/q+r) + b\cos(px/q+r) \leq c) \leftrightarrow 0 \leq c + \sqrt{a^2+b^2})$

$\forall_{abcpqr}(\neg(p = 0) \to \exists_x(x - \text{число}\ \&\ c \leq a\sin(px/q + r) + b\cos(px/q + r)) \leftrightarrow c \leq \sqrt{a^2 + b^2})$

Приемы имеют заголовок "связка". Переменная x идентифицируется с неизвестной задачи на описание, причем все содержащие эту неизвестную условия идентифицируются с подкванторными утверждениями - неравенством и указателем типа значения "$x -$ число". Проверяется, что x не входит в a, b, c, p, q, r. Указатель "альтернатива" разрешает рассмотрение строгих неравенств. Уровень срабатывания равен 1.

Задачи на исследование, имеющие цель "известно"

1. Выражение неизвестного косинуса через синус.

   $\forall_{abcxy}(x = \sin y \to (a - b\cos y)^c(a + b\cos y)^c = (a^2 - b^2 + b^2x^2)^c)$

   Прием имеет заголовок "второйтерм". Произведение входит в посылку задачи на исследование, имеющей цель "известно". Антецедент идентифицируется с другой посылкой. Переменная x является неизвестной для внешней задачи на описание. Выражение y содержит неизвестные, выражения a, b, c - не содержат. Уровень срабатывания равен 2. Цель преобразования - перейти к выражению, содержащему только численные параметры и избавиться от неизвестных тригонометрических операций.

   $\forall_{abABC}(0 \leq a\ \&\ 0 \leq b \to a(\cos(\angle(ABC)))^2 + b = a \leftrightarrow \sqrt{a}\sin(\angle(ABC)) = \sqrt{b})$

   Равенство входит в посылку задачи на исследование либо задачи на доказательство. Выражения a, b не содержат символов "угол", "расстояние", "площадь". Антецеденты обрабатываются проверочными операторами. Уровень срабатывания равен 6.

2. Определение угла при известном косинусе.

   $\forall_{bx}(0 \leq \pi - x\ \&\ \cos x = b \to x = \arccos b)$

   Прием имеет заголовок "вывод". Второй антецедент идентифицируется с посылкой задачи на исследование, имеющей цель "известно". Выражение x содержит неизвестные (не обязательно численные, например, это могут быть переменные для точек чертежа) и встречается в уравнениях с численными неизвестными внешней задачи на описание. Либо это выражение имеет заголовок "угол", либо задача имеет посылку вида $x = \angle(ABC)$. Выражение b не содержит неизвестных. Первый антецедент обрабатывается проверочным оператором. Задача не должна иметь посылку вида $px + q = r$, где выражения p, q, r не содержат отличных от переменных числовых атомов. Уровень срабатывания равен 2.

3. Определение угла при известных синусе и косинусе.

   $\forall_{abx}(0 < x + \pi\ \&\ 0 \leq \pi - x\ \&\ \sin x = a\ \&\ \cos x = b \to x = \mathrm{Sg}(a)\arccos b)$

   Прием имеет заголовок "вывод". Два последних антецедента идентифицируются с посылками задачи на исследование, имеющей цель "известно". Выражение x содержит неизвестные и встречается в уравнениях задачи, содержащих неизвестные внешней задачи на описание. Оно не имеет заголовка "угол", причем отсутствует посылка вида $x = \angle(ABC)$. Выражения a, b не содержат неизвестных. Первые два антецедента обрабатываются проверочными операторами. Уровень срабатывания равен 2.

4. Усмотрение косинуса одной второй пи.

   $\forall_{abc}(a + b + c - \pi = 0 \rightarrow \cos(a/2 + b/2 + c/2) = 0)$

   Прием имеет заголовок "второйтерм". Антецедент берется в контексте преобразуемого выражения. Уровень срабатывания равен 0.

5. Усмотрение противоречивых условий на угол.

   $\forall_{ab}(\cos a = b \;\&\; \pi/2 < a \;\&\; 0 \leq \pi - a \;\&\; 0 \leq b \rightarrow \text{ложь})$

   Прием имеет заголовок "вывод". Первые два антецедента суть посылки задачи на исследование, имеющей цель "контроль". Эта цель появляется при разборе случаев и означает, что посылки могут оказаться противоречивыми. Третий и четвертый антецеденты обрабатываются проверочными операторами. Уровень срабатывания равен 2.

   $\forall_a(0 < a \;\&\; 0 \leq \pi - a \rightarrow \neg(\cos a = 1))$

   $\forall_a(0 < a \;\&\; 0 < \pi - 2a \rightarrow \neg(\cos a \leq 0))$

   Приемы имеют заголовок "второйтерм". Утверждение (равенство либо неравенство) с косинусом входит в условие задачи на описание, имеющей цель "стандравно". Такие задачи используются для решения систем уравнений, извлекаемых из посылок задачи на исследование, имеющей цель "известно". Требуется также, чтобы имел место этап редактирования найденного ответа. Антецеденты обрабатываются проверочными операторами.

6. Выражение косинуса через известный тангенс половинного аргумента.

   $\forall_{abc}(a - 2b = 0 \;\&\; \operatorname{tg} b = c \rightarrow \cos a = (1 - c^2)/(1 + c^2))$

   Прием имеет заголовок "второйтерм". Второй антецедент является посылкой задачи на исследование, имеющей цель "известно". Выражение c не содержит неизвестных. Первый антецедент сравнивает тригонометрические аргументы a и b, он выделен указателем "идентификатор". Уровень срабатывания равен 2.

7. Использование посылки для усмотрения суммы квадратов синуса и косинуса.

   $\forall_{abcdxyz}(a \sin x/\cos y = c \sin z \rightarrow dc^2(\cos z)^2 + da^2(\sin x)^2/(\cos y)^2 = c^2 d)$

   Прием имеет заголовок "второйтерм". Антецедент идентифицируется с посылкой задачи на исследование. Уровень срабатывания равен 2.

8. Вывод следствия с помощью формулы косинуса суммы.

   $\forall_{abcde}(a \cos(b + c) + d = e \rightarrow a \cos b \cos c - a \sin c \sin b + d = e)$

   Прием имеет заголовок "вывод". Антецедент идентифицируется с посылкой задачи на исследование, имеющей цель "известно". Эта посылка не имеет числовых атомов, отличных от переменных. Каждый синус либо косинус, оказывающийся после общей стандартизации в выражениях $\sin b, \sin c, \cos b, \cos c$, встречается в уравнениях задачи. Выражения b, c содержат неизвестные. Уровень срабатывания равен 4.

9. Переход к половинному углу.

   $\forall_{aABC}(a = \angle(ABC) \to \sqrt{1-\cos a} = \sqrt{2}\sin(a/2))$

   Прием имеет заголовок "второйтерм". Антецедент идентифицируется с посылкой задачи на исследование, имеющей цель "известно". Выражение a представляет собой переменную. Уровень срабатывания равен 2.

   $\forall_{ab}(a - 2b = 0 \to \sqrt{1-\cos a} = \sqrt{2}|\sin b|)$

   $\forall_{ab}(a - 2b = 0 \to \sqrt{1+\cos a} = \sqrt{2}|\cos b|)$

   Приемы имеют заголовок "второйтерм". Преобразуемое выражение входит в посылку задачи на исследование, имеющей цель "известно". В первом случае эта посылка имеет также вхождение выражения $\sin b$, во втором случае - выражения $\cos b$. Антецедент, выделенный указателем "идентификатор", сравнивает углы a, b. Уровень срабатывания равен 2.

10. Использование посылки для суммы трех углов треугольника.

    $\forall_{xyz}(x + y + z = \pi \to \cos(x+y) = -\cos z)$

    Прием имеет заголовок "второйтерм". Косинус входит в уравнение - посылку задачи на исследование, имеющей цель "известно". Антецедент идентифицируется с другой посылкой. Выражение z встречается в текущем уравнении. Уровень срабатывания равен 4.

11. Вывод следствия о знаке косинуса при контроле непротиворечивости подслучая.

    $\forall_{ab}(\cos b = a \ \& \ 0 \leq b \ \& \ 0 < \pi/2 - b \to 0 < a)$

    Первый антецедент идентифицируется с посылкой задачи на исследование, имеющей цель "контроль". Переменная b является неизвестной, выражение a неконстантное и не содержит неизвестных. Второй и третий антецеденты обрабатываются проверочными операторами. Уровень срабатывания равен 1.

12. Усмотрение неотрицательности косинуса.

    $\forall_{abx}(a\cos x = b \ \& \ 0 < a \ \& \ 0 < b \to 0 < \cos x)$

    Прием имеет заголовок "вывод". Первый антецедент идентифицируется с посылкой задачи на исследование, имеющей цель "известно". Эта посылка должна содержать символ "длина". Второй и третий антецеденты обрабатываются проверочными операторами. Уровень срабатывания равен 3. Прием используется в задачах по физике. Потребность в нем и ряде аналогичных приемов возникает из-за того, что в процессе преобразований утверждения, позволяющие легко усмотреть знак параметра, могут оказаться утраченными, и усмотрение лучше делать заблаговременно.

13. Определение косинуса суммы по синусу и косинусу угла.

    $\forall_{ABCabc}(\sin(\angle(ABC)) = a \ \& \ \cos(\angle(ABC)) = b \to \cos(\angle(ABC) + c) = b\cos c - a\sin c)$

    Прием имеет заголовок "второйтерм". Косинус входит в посылку задачи на исследование, имеющей цель "известно". Антецеденты идентифицируются с другими посылками. Выражения a, b, c не содержат неизвестных. Уровень срабатывания равен 2.

Нормализатор общей стандартизации "нормкосинус"

В нормализаторе представлены следующие приемы, копирующие ранее рассмотренные приемы общей стандартизации:

1. Косинус константы: $0, \pi/2, \pi/6, \pi/4, \pi/3$.
2. Косинус целого кратного пи.
3. Косинус целого кратного половины пи.
4. Минус под косинусом.
5. Модуль под косинусом.
6. Аргумент домножен на степень минус единицы.
7. Преобразование дробного слагаемого аргумента к виду суммы дробей.
8. Изменение знака разности под косинусом.
9. Формулы приведения.
10. Косинус арккосинуса.
11. Косинус арксинуса.
12. Косинус арктангенса.
13. Косинус половины арккосинуса.

11.3 Приемы символа "тангенс"

Выражение "тангенс(a)", прорисовываемое формульным редактором в виде $\operatorname{tg} a$, обозначает тангенс вещественного числа a. Справочники "тип", "арность", "одз", "типданных" характеризуют простейшие свойства символа "тангенс".

Общая стандартизация выражений

1. Тангенс константы.
    (a) Тангенс нуля.
    $\operatorname{tg} 0 = 0$
    (b) Тангенс одной шестой пи.
    $\operatorname{tg}(\pi/6) = 1/\sqrt{3}$
    (c) Тангенс одной четвертой пи.
    $\operatorname{tg}(\pi/4) = 1$
    (d) Тангенс одной третьей пи.
    $\operatorname{tg}(\pi/3) = \sqrt{3}$

    Уровень срабатывания перечисленных приемов равен 0.

  (e) Тангенс одной восьмой пи.

  $\mathrm{tg}(\pi/8) = \sqrt{2} - 1$

  Для условия задачи на преобразование, имеющей цель "известны" (т.е. при редактировании ответа задачи на вычисление), уровень срабатывания равен 3. В остальных случаях он равен 6. В задачах на исследование, имеющих цель "известно", прием не применяется.

  $\forall_{ab}(a\cos(\pi/8)/b\sin(\pi/8) = a/(\sqrt{2}-1)b)$

  $\forall_{ab}(a\sin(\pi/8)/b\cos(\pi/8) = (\sqrt{2}-1)a/b)$

  Уровень срабатывания равен 1.

  (f) Тангенс одной двенадцатой пи.

  $\mathrm{tg}(\pi/12) = 2 - \sqrt{3}$

  Аналогично тангенсу одной восьмой пи.

2. Минус под тангенсом.

   $\forall_a(\mathrm{tg}(-a) = -\mathrm{tg}\, a)$

   Уровень срабатывания равен 0.

3. Тангенс целого кратного пи.

   $\forall_n(n - \text{целое} \rightarrow \mathrm{tg}(\pi n) = 0)$

   Антецедент обрабатывается проверочным оператором. Введен ускоряющий указатель "логсимвол(пи умножение)", определяющий идентификацию символа "пи" как непосредственного операнда произведения. Допускается вырожденное единичное значение n. Уровень срабатывания равен 1.

4. Преобразование дробного слагаемого аргумента к виду суммы дробей.

   $\forall_{abcdefn}(f = (a+b)^n c/d+e \;\&\; \neg(\cos((a+b)^n c/d+e) = 0) \rightarrow \mathrm{tg}((a+b)^n c/d+e) = \mathrm{tg}\, f)$

   Допускаются вырожденные единичные значения c, d, n, однако либо c, либо n должно быть отлично от единицы. Допускается также нулевое значение e. Показатель степени n идентифицируется с натуральной константой, не превосходящей 3. Если выражение c содержит неизвестные задачи, а $a + b$ - не содержит, то прием блокируется. Первый антецедент выделен указателем "идентификатор". Для раскрывания скобок используется обращение к нормализатору "стандплюс". Второй антецедент обрабатывается проверочным оператором. Он проверяет условия на о.д.з. для тангенса и передает ссылки на утверждения контекста задачи, из которых эти условия вытекают, новому сопровождающему утверждению $\neg(\cos f = 0)$. Уровень срабатывания равен 0. В особых случаях переход к сумме выполняется без проверки того, что косинус не равен 0:

   $\forall_{abcdn}(n - \text{целое} \rightarrow \mathrm{tg}((an+b)\pi c/d) = \mathrm{tg}(ac\pi n/d + b\pi c/d))$

   Здесь n - переменная. Антецедент обрабатывается проверочным оператором. Никаких дополнительных ограничений на срабатывание нет. Уровень срабатывания равен 2.

5. Формулы приведения.

   $\forall_{abcde}(e = bc + d \;\&\; 0 < c \rightarrow \mathrm{tg}(a + \pi e/b) = \mathrm{tg}(a + \pi d/b))$

Прием делает положительный рациональный коэффициент при π меньшим единицы. b, e - натуральные константы. Антецеденты выделены указателем "программа". Уровень срабатывания равен 0.

$\forall_{abcd}(c = 2a - b \ \& \ 0 \leq c \rightarrow \operatorname{tg}(d + \pi a/b) = -\operatorname{ctg}(d + \pi c/2b))$

Коэффициент при π делается меньшим одной второй. a, b - натуральные константы. Уровень срабатывания равен 0.

$\forall_{abcdef}(a = bc + e \ \& \ f = b - e \rightarrow \operatorname{tg}(d - \pi a/b) = \operatorname{tg}(d + \pi f/b))$

Прием переходит от отрицательно коэффициента при π к положительному. a, b - натуральные константы. Уровень срабатывания равен 0.

$\forall_{abcde}(c = 4e - b \ \& \ 0 \leq c \ \& \ d = b - 2e \rightarrow \operatorname{tg}(a + \pi e/b) = \operatorname{ctg}(-a + \pi d/2b))$

Прием делает коэффициент при π не превосходящим одной четвертой. Для коэффициента, равного одной четвертой, замена выполняется, если либо a имеет заголовок "минус", либо обработанный нормализатором терм $-a$ лексикографически предшествует терму a. Уровень срабатывания равен 0.

$\forall_{an}(n - \text{целое} \rightarrow \operatorname{tg}(a + \pi n) = \operatorname{tg} a)$

Антецедент обрабатывается проверочным оператором. Уровень срабатывания равен 0.

6. Произведение тангенса на косинус.

   $\forall_{abcdef}(a = \min(b, c) \ \& \ d = b - a \ \& \ e = c - a \rightarrow (\operatorname{tg} f)^b(\cos f)^c = (\operatorname{tg} f)^d(\sin f)^a(\cos f)^e)$

   Переменные b, c идентифицируются с натуральными константами. Допускаются вырожденные единичные значения. Антецеденты выделены указателем "программа". Уровень срабатывания равен 3.

   $\forall_{ab}(\operatorname{tg}(a - b)\cos(b - a) = \sin(a - b))$

   Уровень срабатывания равен 1.

   $\forall_{ab}(0 \leq \cos b \rightarrow \sqrt{a \operatorname{tg} b}\cos b = \sqrt{a\sin(2b)/2})$

   Антецедент обрабатывается проверочным оператором. Введен сравнительно сильный ограничитель трудоемкости. Уровень срабатывания равен 2.

7. Деление тангенса на синус.

   $\forall_{abc}(a \operatorname{tg} b/c \sin b = a/c \cos b)$

   Уровень срабатывания равен 0.

8. Сумма квадрата тангенса и единицы.

   $\forall_{abc}(a + a(\operatorname{tg} b)^2 + c = a/(\cos b)^2 + c)$

   Преобразование применяется только в посылках задач на доказательство либо исследование. В других случаях будет применяться аналогичный прием нормализатора "видумножение". Уровень срабатывания равен 1.

9. Частичное сокращение для синуса и тангенса.

   $\forall_{abcn}(a(\sin b)^n/c(\operatorname{tg} b)^n = a(\cos b)^n/c)$

   Указатель "дробь" разрешает перестановку числителя и знаменателя. Уровень срабатывания равен 1.

10. Сокращение тангенсов аргументов, отличающихся знаком.

    $\forall_{abcd}(\neg(\operatorname{tg}(c-b)=0) \to a\operatorname{tg}(b-c)/d\operatorname{tg}(c-b) = -a/d)$

    Антецедент обрабатывается проверочным оператором. Уровень срабатывания равен 1.

11. Преобразование подынтегрального выражения. Единственный прием этого подраздела связан с произведением тангенсов:

    $\forall_{abc}(\operatorname{tg}(a+c)\operatorname{tg}(b+c) = (\operatorname{tg}(a+c)-\operatorname{tg}(b+c))/\operatorname{tg}(a-b)-1)$

    Решается вспомогательная задача на преобразование подынтегрального выражения, имеющая цель "нормИнтеграл". Переменная интегрирования x входит в выражение c и не входит в выражения a, b. Допустимые заголовки надвыражений преобразуемого произведения суть "плюс", "минус", "умножение", "модуль", "дробь". Это произведение не расположено внутри дроби, знаменатель которой зависит от x. Для сопровождения по о.д.з. выводится утверждение $\neg(\operatorname{tg}(a-b)=0)$. Уровень срабатывания равен 2.

12. Редактирование ответа задачи на вычисление.

    (a) Сокращение для тангенса и синуса двойного аргумента.

       $\forall_{abpq}(p = 2q \to a\sin p/b\operatorname{tg} q = 2a(\cos q)^2/b)$

       Прием применяется в задачах на преобразование, имеющих цель "известны". Антецедент сравнивает тригонометрические аргументы и выделен указателем "идентификатор". Уровень срабатывания равен 2.

    (b) Сокращение на тангенс с переходом к котангенсу.

       $\forall_{abcde}(\neg(\sin b = 0) \to a\operatorname{tg} b/(c(d\operatorname{tg} b + e)) = a/(c(d + e\operatorname{ctg} b)))$

       Прием применяется в задачах на преобразование, имеющих цель "учетрезультата" либо "известны". Антецедент обрабатывается проверочным оператором. Уровень срабатывания равен 3.

    (c) Сумма тангенсов двух углов треугольника.

       $\forall_{ABCabc}(\angle(ABC) = b \;\&\; \angle(ACB) = c \;\&\; \angle(BAC) = a \to \operatorname{tg} a + \operatorname{tg} b = \sin c/(\cos a\cos b))$

       Прием применяется в задачах на преобразование, имеющих цель "известны". Выражения a, b, c не содержат неизвестных. Уровень срабатывания равен 1.

13. Произведение квадратного трехчлена с тангенсом на квадрат косинуса.

    $\forall_{abx}((\cos x)^2(a - a(\operatorname{tg} x)^2 + b\operatorname{tg} x) = a\cos(2x) + b\sin(2x)/2)$

    Преобразуемое выражение входит в условие задачи на преобразование. Уровень срабатывания равен 2.

Усмотрение тангенса суммы

$\forall_{abcdef}(\neg(d - d\operatorname{tg} a\operatorname{tg} b = 0) \;\&\; \neg(\cos a = 0) \;\&\; \neg(\cos b = 0) \to e(\operatorname{tg} a + \operatorname{tg} b)^c/f(d - d\operatorname{tg} a\operatorname{tg} b)^c = e(\operatorname{tg}(a+b))^c/fd^c)$

Дробь входит в условие задачи на преобразование. Тригонометрические аргументы a, b не имеют вхождений в условие, расположенных вне данной дроби. Для сопровождения по о.д.з. выводится утверждение $\neg(\cos(a+b)=0)$. Уровень срабатывания равен 3. Если под тангенсами находятся константы, то применяется другая версия приема:

$\forall_{abcdefg}(c -$ натуральное $\& \neg(d=0) \& \neg(\cos g=0) \& g=a+b \rightarrow e(\operatorname{tg} a+\operatorname{tg} b)^c/f(d-d\operatorname{tg} a\operatorname{tg} b)^c = e(\operatorname{tg} g)^c/fd^c)$

Переменные a, b идентифицируются с десятичными константами. Первые три антецедента обрабатываются проверочными операторами, четвертый - выделен указателем "программа". Уровень срабатывания равен 0. Для особого случая обращения тангенса в единицу созданы еще два приема:

$\forall_{abcde}(\neg(b-b\operatorname{tg} a=0) \& \neg(\cos a=0) \rightarrow d(\operatorname{tg} a+1)^c/e(b-b\operatorname{tg} a)^c = d(\operatorname{tg}(a+\pi/4))^c/eb^c)$

$\forall_{abcde}(\neg(b-b\operatorname{ctg} a=0) \& \neg(\sin a=0) \rightarrow d(\operatorname{ctg} a+1)^c/e(b-b\operatorname{ctg} a)^c = d(-\operatorname{tg}(a+\pi/4))^c/eb^c)$

Дробь входит в условие задачи на преобразование. Тригонометрический аргумент a не имеет вхождений в условие, расположенных вне данной дроби. Антецеденты обрабатываются проверочными операторами. Для сопровождения по о.д.з. выводится утверждение $\neg(\cos(a+\pi/4)=0)$. Уровень срабатывания равен 3.

Выражение тангенса через синус и косинус

Обычно переход от тангенса к отношению синуса и косинуса выполняется нормализатором "видумножение" либо приводимым далее приемом преобразования уравнений. В особых случаях применяются дополнительные три приема, основанные на следующей теореме:

$\forall_a(\operatorname{tg} a=\sin a/\cos a)$

Первый прием срабатывает в условии задачи на преобразование, имеющей цель "упростить", если это условие - константное и имеет тригонометрический аргумент, отличный от a. Уровень срабатывания равен 4.

Второй прием срабатывает в условии задачи на преобразование, имеющей цель "нормИнтеграл", т.е. при предварительном упрощении подынтегрального выражения. Переменная интегрирования x входит в a. Тангенс не должен располагаться внутри выражения с заголовком "логарифм". Если тангенс является обобщенным сомножителем подынтегрального выражения и расположен внутри степени с дробным показателем, то должен найтись содержащий x тригонометрический аргумент, либо отличный от a, либо не расположенный под тангенсом или котангенсом. Уровень срабатывания тоже равен 4.

Наконец, третий прием применяется в условии задачи на доказательство неравенства. Уровень его срабатывания равен 6.

Равенство тангенса нулю

$\forall_a(\operatorname{tg} a=0 \leftrightarrow \sin a=0)$

Уровень срабатывания равен 2.

Тангенс выражения, содержащего обратные тригонометрические функции

1. Тангенс арксинуса.

   $\forall_a (0 < 1 + a \ \& \ 0 < 1 - a \to \operatorname{tg} \arcsin a = a/\sqrt{1 - a^2})$

   Антецеденты обрабатываются проверочными операторами. Для сопровождения по о.д.з. выводится неравенство $0 < 1 - a^2$. Если выражение a константное, то уровень срабатывания равен 0, иначе он равен 1. В случае посылки задачи на доказательство либо на исследование, а также в случае задачи на преобразование, имеющей цель "известны", используется другая версия приема, с отброшенными антецедентами:

   $\forall_a (\operatorname{tg} \arcsin a = a/\sqrt{1 - a^2})$

   Уровень срабатывания этой версии равен 0.

2. Тангенс арккосинуса.

   $\forall_a (\neg(a = 0) \ \& \ 0 \le 1 + a \ \& \ 0 \le 1 - a \to \operatorname{tg} \arccos a = \sqrt{1 - a^2}/a)$

   Антецеденты обрабатываются проверочными операторами. Для сопровождения по о.д.з. выводится неравенство $0 \le 1 - a^2$. Уровень срабатывания равен 1.

3. Тангенс арктангенса.

   $\forall_a (\operatorname{tg} \operatorname{arctg} a = a)$

   Уровень срабатывания равен 0.

4. Тангенс половины арксинуса.

   $\forall_a (\neg(a = 0) \ \& \ 0 \le 1 + a \ \& \ 0 \le 1 - a \to \operatorname{tg}(\arcsin a/2) = (1 - \sqrt{1 - a^2})/a)$

   Для сопровождения по о.д.з. выводится неравенство $0 \le 1 - a^2$. Уровень срабатывания равен 2.

5. Тангенс половины арккосинуса.

   $\forall_a (\operatorname{tg}(\arccos a/2) = \sqrt{(1 - a)/(1 + a)})$

   Выражение входит в условие задачи. Уровень срабатывания равен 2.

6. Тангенс половины арктангенса.

   $\forall_a (\neg(a = 0) \to \operatorname{tg}(\operatorname{arctg} a/2) = (\sqrt{a^2 + 1} - 1)/a)$

   Аналогично предыдущему.

7. Тангенс удвоенного арксинуса.

   $\forall_a (\neg(1 - 2a^2 = 0) \ \& \ 0 \le 1 - a \ \& \ 0 \le 1 + a \to \operatorname{tg}(2 \arcsin a) = 2a\sqrt{1 - a^2}/(1 - 2a^2))$

   Выражение входит в условие задачи. Антецеденты обрабатываются проверочными операторами. Уровень срабатывания равен 2. Для задач на преобразование, имеющих цель "известны", создана версия приема с отброшенными антецедентами.

8. Тангенс удвоенного арккосинуса.

   $\forall_a (\neg(2a^2 - 1 = 0) \ \& \ 0 \le 1 + a \ \& \ 0 \le 1 - a \to \operatorname{tg}(2 \arccos a) = 2a\sqrt{1 - a^2}/(2a^2 - 1))$

   Аналогично предыдущему.

9. Тангенс удвоенного арктангенса.

   $\forall_a(\neg(1-a^2=0) \to \mathrm{tg}(2\,\mathrm{arctg}\,a)=2a/(1-a^2))$

   Аналогично предыдущему.

10. Использование формулы тангенса суммы.

    $\forall_{ab}(\neg(\cos a=0)\ \&\ \neg(\cos b=0)\ \&\ \neg(\cos(a+b)=0) \to \mathrm{tg}(a+b)=(\mathrm{tg}\,a+\mathrm{tg}\,b)/(1-\mathrm{tg}\,a\,\mathrm{tg}\,b))$

    Тангенс входит в условие задачи на преобразование либо в условие задачи на описание, для которой имеет место этап редактирования ответа. Хотя бы одно из выражений a, b содержит обратную тригонометрическую функцию. Антецеденты обрабатываются проверочными операторами. Для сопровождения по о.д.з. выводится утверждение $\neg(1-\mathrm{tg}\,a\,\mathrm{tg}\,b=0)$. Уровень срабатывания приема равен 2.

11. Обращение к вспомогательной задаче для преобразования тангенса от выражения, содержащего обратную тригонометрическую функцию.

    $\forall_{ab}(a=\mathrm{tg}\,b \to \mathrm{tg}\,b=a)$

    Аналогично случаям синуса и косинуса.

Уравнения

1. Простейшее уравнение для тангенса.

   $\forall_{ab}(\mathrm{tg}\,a=b \leftrightarrow \exists_n(n-\text{целое}\ \&\ a=\mathrm{arctg}\,b+\pi n))$

   Уравнение является условием задачи на описание, не имеющей цели "известно". Выражение a содержит неизвестные, b - не содержит. Оператор "фильтрсерии" ложен. Если имеется условие $\neg(\cos a=0)$, которое далее не нужно для сопровождения по о.д.з., то оно отбрасывается. Преобразованное уравнение снабжается комментарием "серия". Уровень срабатывания равен 2.

   $\forall_{ab}(0\le a\ \&\ 0\le \pi-a\ \&\ 0\le b \to \mathrm{tg}\,a=b \leftrightarrow a=\mathrm{arctg}\,b)$

   $\forall_{ab}(0\le a\ \&\ 0\le \pi-a\ \&\ b\le 0 \to \mathrm{tg}\,a=b \leftrightarrow a=\pi+\mathrm{arctg}\,b)$

   $\forall_{ab}(a\le 0\ \&\ 0\le \pi+a\ \&\ b\le 0 \to \mathrm{tg}\,a=b \leftrightarrow a=\mathrm{arctg}\,b)$

   Уравнение является условием задачи на описание, имеющей цель "стандравно" (решение вспомогательных уравнений, извлеченных из планиметрической задачи). Выражение a содержит неизвестные, выражение b - не содержит. Как и выше, отбрасывается не используемое для сопровождения по о.д.з. условие $\neg(\cos a=0)$. Уровень срабатывания равен 0.

   $\forall_{ab}(0\le a\ \&\ 0\le \pi-a \to \mathrm{tg}\,a=b \leftrightarrow 0\le b\ \&\ a\,\mathrm{arctg}\,b\ \vee\ b<0\ \&\ a=\pi+\mathrm{arctg}\,b)$

   Уровень срабатывания равен 1.

2. Усмотрение уравнения относительно тангенса.

   $\forall_{abcde}(a\sin b+c\cos b+d(\sin b)^2\cos b+e(\cos b)^3=0 \leftrightarrow \neg(\cos b=0)\ \&\ a\,\mathrm{tg}\,b+c+(d(\mathrm{tg}\,b)^2+e)/(1+(\mathrm{tg}\,b)^2)=0\ \vee\ \cos b=0\ \vee\ a=0)$

   Уравнение является условием задачи на описание. Выражение b содержит неизвестные, выражения a, c, d, e не содержат неизвестных. Указатели "подстановка" разрешают обращение в 0 коэффициентов d, e. Уровень срабатывания равен 5.

$\forall_{abx}(a\sin x/b\cos x = a\operatorname{tg} x/b)$

Прием выполняет тождественную замену, приводящую к уравнению с тангенсами. Дробь входит в условие задачи на описание. Переменная x идентифицируется с неизвестной. Любое вхождение ее в текущее условие вне данной дроби должно быть расположено непосредственно под тангенсом или котангенсом. Текущее условие не содержит отличных от x неизвестных тригонометрических аргументов. Уровень срабатывания равен 1.

3. Переход к тангенсу половинного угла.

   Переход к тангенсу половинного угла предпринимается на достаточно высоком уровне - если прочие средства не позволили решить задачу. Для этого необходимо усмотрение одновременно синуса и косинуса общего неизвестного аргумента, либо лишь одной из этих операций вместе с уже имеющимся тангенсом половинного аргумента.

   $\forall_{abcde}(c = a \mathbin{\&} e = d/2 \rightarrow a = b \leftrightarrow \neg(\cos e = 0) \mathbin{\&} c = b \ \vee\ \cos e = 0 \mathbin{\&} a = b)$

   На этой теореме созданы два приема, инициируемые при усмотрении косинуса и синуса соответственно. Уравнение $a = b$ является условием задачи на описание. a содержит неизвестные, b - не содержит. Ограничимся рассмотрением первого приема, так как второй получается из него простой перестановкой символов "синус" и "косинус". Указатель "контекст(позиция(x7 x1)вид(x7 косинус(x4)))" определяет усмотрение внутри левой части уравнения выражения $\cos d$. Проверяется, что d содержит неизвестные. Антецеденты выделены указателями "идентификатор". Первый из них обращается к нормализатору "половинныйугол", который собственно и выполняет переход к половинному углу (см. ниже). Второй определяет этот половинный угол e. Уравнение должно содержать либо выражение $\sin d$, либо выражение $\operatorname{tg} e$. Оно не содержит тригонометрических операций с неизвестными, отличных от $\sin d, \cos d, \operatorname{tg} e, \operatorname{tg} d$. Если такая тригонометрическая операция находится в основании степени, то показателем степени должна быть константа 2 либо 3. При работе с комплексными числами и при решении дифференциальных уравнений прием блокируется. Проверяется, что после применения нормализатора "половинныйугол" не остается вхождений тригонометрического аргумента d под синусом либо косинусом. Уровень срабатывания равен 6.

4. Нормализатор "половинныйугол". Нормализатор является корневым, т.е. перечисляемые ниже преобразования применяются только к корню выражения, а не к его подвыражениям. При обращении к нормализатору задается комментарий (угол a), где a - угол, от которого нужно перейти к половинному углу.

   (a) Выражение синуса через тангенс половинного угла.
   $\forall_a(\sin a = 2\operatorname{tg}(a/2)/(1 + (\operatorname{tg}(a/2))^2))$
   Аргумент a берется из комментария (угол a).

   (b) Выражение косинуса через тангенс половинного угла.
   $\forall_a(\cos a = (1 - (\operatorname{tg}(a/2))^2)/(1 + (\operatorname{tg}(a/2))^2))$

   (c) Выражение тангенса через тангенс половинного угла.
   $\forall_{ab}(\neg(\cos a = 0) \mathbin{\&} b = a/2 \rightarrow \operatorname{tg} a = 2\operatorname{tg} b/(1 - (\operatorname{tg} b)^2))$

Первый антецедент обрабатывается проверочным оператором, второй - выделен указателем "идентификатор". Аргумент a берется из комментария (угол a). Для сопровождения по о.д.з. выводится утверждение $\neg(1 - (\operatorname{tg} b)^2 = 0)$.

(d) Сумма выражений.

Для каждого слагаемого суммы предпринимается независимое рекурсивное обращение к нормализатору "половинныйугол":

$\forall_{abc}(c = a \to a + b = b + c)$

Антецедент выделен указателем "идентификатор", его правая часть обрабатывается нормализатором "половинныйугол". Проверяется, что результат отличен от исходного выражения.

(e) Выражение с заголовком "минус".

$\forall_{ab}(b = a \to -a = -b)$

Рекурсивное обращение, аналогично предыдущему.

(f) Произведение выражений.

$\forall_{abc}(c = a \to ab = bc)$

Рекурсивное обращение.

(g) Дробное выражение. Реализуются рекурсивные обращения:

$\forall_{abc}(c = a \to a/b = c/b)$

$\forall_{abc}(\neg(b = 0)\ \&\ c = b \to a/b = a/c)$

Во втором приеме первый антецедент обрабатывается проверочным оператором, второй - выделен указателем "идентификатор". Для сопровождения по о.д.з. выводится утверждение $\neg(c = 0)$.

(h) Степенное выражение. Реализуются рекурсивные обращения:

$\forall_{abc}(0 < a\ \&\ c = b \to a^b = a^c)$

Выражение b неконстантное.

$\forall_{abc}(b - \mathrm{rational}\ \&\ \neg(\text{знаменатель}(b) - \mathrm{even})\ \&\ c = a \to a^b = c^b)$

(i) Сложение дробей с одинаковым знаменателем. Для предварительного упрощения введен прием, складывающий дроби с одинаковым знаменателем:

$\forall_{abcd}((a + b)/c + d = a/c + b/c + d)$

Направление замены - справа налево. Допускается вырожденное нулевое d.

5. Замена отрицания равенства нулю тангенса на отрицание равенства нулю синуса.

$\forall_a(\neg(\cos a = 0) \to \neg(\operatorname{tg} a = 0) \leftrightarrow \neg(\sin a = 0))$

Преобразование применяется даже к утверждениям, используемым для сопровождения по о.д.з. Уровень срабатывания равен 1.

6. Деление членов уравнения на косинус для перехода к уравнению с тангенсом.

$\forall_{abc}(a \cos b + c \sin b = 0 \leftrightarrow \neg(\cos b = 0)\ \&\ a + c \operatorname{tg} b = 0\ \vee\ \cos b = 0\ \&\ c = 0)$

Уравнение является условием задачи на описание, имеющей единственную неизвестную. Выражение b содержит неизвестные. Выражения a, c не имеют тригонометрического аргумента, отличного от $2b$ и b, причем все вхождения тригонометрического аргумента b в эти выражения находятся непосредственно под тангенсом. Указатели "группировка" определяют идентификацию выражений a, c путем группировки всех слагаемых, делящихся, соответственно, на $\cos b$ и $\sin b$. Уровень срабатывания равен 5.

7. Выражение тангенса через синус и косинус.

   $\forall_a(\operatorname{tg} a = \sin a/\cos a)$

   Тангенс входит в условие задачи на описание, имеющей более одной неизвестной. Выражение a содержит неизвестные. Должно существовать уравнение задачи, содержащее $\sin a$ либо $\cos a$. Уровень срабатывания равен 2.

8. Тангенс суммы.

   $\forall_{ab}(\neg(\cos a = 0)\ \&\ \neg(\cos b = 0)\ \&\ \neg(\cos(a+b) = 0) \to \operatorname{tg}(a+b) = (\operatorname{tg} a + \operatorname{tg} b)/(1 - \operatorname{tg} a \operatorname{tg} b))$

   Тангенс входит в условие задачи на описание, имеющей более одной неизвестной. Выражение a содержит неизвестные. Существует уравнение задачи, содержащее $\operatorname{tg} a$ либо $\operatorname{ctg} a$. Для сопровождения по о.д.з. выводится утверждение $\neg(1 - \operatorname{tg} a \operatorname{tg} b = 0)$. Уровень срабатывания равен 2.

9. Существование решения уравнения.

   $\forall_a(\exists_x(x - \text{число}\ \&\ a = \operatorname{tg} x\ \&\ \neg(\cos x = 0)) \leftrightarrow a - \text{число})$

   Прием имеет заголовок "связка". Существование тангенса, равного a, эквивалентно условию a – число. Уровень срабатывания равен 0.

Неравенства

1. Решение простейших нестрогих неравенств с тангенсом.

   $\forall_{ab}(\neg(\cos b = 0) \to \operatorname{tg} b \leq a \leftrightarrow \exists_n(n - \text{целое}\ \&\ \pi n - \pi/2 < b\ \&\ b \leq \pi n + \operatorname{arctg} a))$

   $\forall_{ab}(\neg(\cos b = 0) \to a \leq \operatorname{tg} b \leftrightarrow \exists_n(n - \text{целое}\ \&\ \pi n + \operatorname{arctg} a \leq b\ \&\ b < \pi n + \pi/2))$

   Неравенство является условием задачи на описание, не используемым для сопровождения по о.д.з. Задача не имеет уравнений. Выражение b содержит неизвестные, a - не содержит. Оператор "фильтрпромежутков" ложен. Преобразованное условие снабжается комментарием "серия". Если имеется условие вида $\neg(\cos b = 0)$, не используемое далее для сопровождения по о.д.з., то оно удаляется. Уровни срабатывания равны 2 и 6.

2. Решение простейших строгих неравенств с тангенсом.

   $\forall_{ab}(\neg(\cos b = 0) \to \operatorname{tg} b < a \leftrightarrow \exists_n(n - \text{целое}\ \&\ \pi n - \pi/2 < b\ \&\ b < \pi n + \operatorname{arctg} a))$

   $\forall_{ab}(\neg(\cos b = 0) \to a < \operatorname{tg} b \leftrightarrow \exists_n(n - \text{целое}\ \&\ \pi n + \operatorname{arctg} a < b\ \&\ b < \pi n + \pi/2))$

   Аналогично предыдущему.

3. Решение двусторонних неравенств с тангенсами.

   $\forall_{abc}(\neg(\cos b = 0) \to a \leq \operatorname{tg} b\ \&\ \operatorname{tg} b \leq c \leftrightarrow \exists_n(n - \text{целое}\ \&\ \pi n + \operatorname{arctg} a \leq b\ \&\ b \leq \pi n + \operatorname{arctg} c))$

   Прием имеет заголовок "заменаусловия(второйтерм)". Неравенства представляют собой условия задачи на описание, не используемые для сопровождения по о.д.з. Задача не имеет уравнений. Выражение b содержит неизвестные, выражения a, c - не содержат. Отсутствуют другие неравенства с $\operatorname{tg} b$ в качестве одной из своих частей. Отсутствует условие вида $\neg(\operatorname{tg} b = d)$. Оператор "фильтрпромежутков" ложен. Указатели "альтернатива" позволяют учесть случаи, когда оба неравенства или одно из них строгие. По мере возможности, удаляется условие $\neg(\cos b = 0)$. Новое условие сопровождается комментарием "серия". Уровни срабатывания равны 1 и 5.

4. Переход к тангенсу половинного угла.

   $\forall_{abcde}(c = a \ \& \ e = d/2 \rightarrow a < b \leftrightarrow \neg(\cos e = 0) \ \& \ c < b \ \vee \ \cos e = 0 \ \& \ a < b)$

   $\forall_{abcde}(c = a \ \& \ e = d/2 \rightarrow a \leq b \leftrightarrow \neg(\cos e = 0) \ \& \ c \leq b \ \vee \ \cos e = 0 \ \& \ a \leq b)$

   Аналогично случаю уравнений. Для каждой из двух теорем созданы по две версии приема. Одна версия инициируется усмотрением $\sin d$, другая - усмотрением $\cos d$. В отличие от уравнений, приемы имеют два уровня срабатывания, 2 и 6. Уровень 2 применяется, если неизвестная часть неравенства представляет собой линейную комбинацию синуса и косинуса, взятых с известными коэффициентами.

5. Преобразование отрицания равенства с неизвестным тангенсом в дизъюнкцию двух неравенств.

   $\forall_{ax}(\neg(\operatorname{tg} x = a) \leftrightarrow \operatorname{tg} x < a \ \vee \ a < \operatorname{tg} x)$

   Отрицание равенства является условием задачи на описание, не имеющей уравнений. Существует неравенство с тригонометрической операцией от неизвестной в одной части и известным выражением в другой. Выражение x содержит неизвестные, a - не содержит. Преобразованное условие снабжается комментарием "разборслучаев". Уровень срабатывания равен 1.

6. Существование решения неравенств.

   $\forall_a(a - \text{число} \rightarrow \exists_x(a < \operatorname{tg} x \ \& \ \neg(\cos x = 0) \ \& \ x - \text{число}))$

   Прием имеет заголовок "связка". Указатели "дробь", "альтернатива" разрешают переставлять части неравенства и переходить к нестрогому неравенству. Уровень срабатывания равен 1.

Задачи на исследование, имеющие цель "известно"

1. Выражение тангенса через известный тангенс половинного аргумента.

   $\forall_{abc}(a - 2b = 0 \ \& \ \operatorname{tg} b = c \rightarrow \operatorname{tg} a = 2c/(1 - c^2))$

   Прием имеет заголовок "второйтерм". Преобразуемый тангенс входит в посылку задачи на исследование, имеющей цель "известно". Второй антецедент является другой посылкой этой задачи. Выражение c не содержит неизвестных. Первый антецедент, выделенный указателем "идентификатор", сравнивает тригонометрические аргументы a и b. Уровень срабатывания равен 2.

2. Произведение тангенса и синуса двойного аргумента.

   $\forall_{ab}(2a - b = 0 \rightarrow \sin b \operatorname{tg} a = 1 - \cos b)$

   Прием имеет заголовок "второйтерм". Произведение входит в посылку задачи на исследование либо на доказательство. Антецедент выделен указателем "идентификатор". Уровень срабатывания равен 3.

3. Сокращение уравнения с тангенсом.

   $\forall_{abc}(a\sin c = b\operatorname{tg} c \leftrightarrow a\cos c = b)$

   Уравнение входит в посылку задачи на исследование, имеющей цель "известно". Уровень срабатывания равен 1. Разумеется, прием мог бы быть полезен в любых задачах, однако обычно вне задач на исследование предпринимается группировка неизвестных членов уравнения в левой части, и рассматриваемая в приеме ситуация там возникать не будет.

4. Усмотрение значения тангенса из соотношения пропорциональности для синуса и косинуса.

   $\forall_{abx}(\neg(a=0) \to a\sin x = b\cos x \leftrightarrow \operatorname{tg} x = b/a)$

   Уравнение входит в посылку задачи на исследование, имеющей цель "известно". Выражение x содержит неизвестные, выражения a, b - не содержат. Уровень срабатывания равен 2.

5. Усмотрение косинуса двойного угла из соотношения пропорциональности для синусов двух углов и уравнения для произведения их тангенсов.

   $\forall_{abcdxyAB}(A = b^2/a^2 \ \& \ B = d^2/c^2 \ \& \ \neg(B-1=0) \ \& \ \neg(a=0) \ \& \ \neg(c=0) \ \& \ a\sin x = b\sin y \to c\operatorname{tg} x\operatorname{tg} y = d \leftrightarrow \cos(2y) = (\sqrt{B(BA^2-2AB+B+4A)} - A - B)/A(B-1))$

   Уравнение входит в посылку задачи на исследование, имеющей цель "известно". Выражения x, y содержат неизвестные, выражения a, b, c, d - не содержат. Последний антецедент идентифицируется с другой посылкой задачи. Первые два антецедента выделены указателем "идентификатор". К их правым частям применяется нормализатор "видумножение". Следующие три антецедента обрабатываются проверочными операторами. Указатель "теквхожд(6)" означает, что прием инициируется усмотрением последнего антецедента, а преобразуемое уравнение затем находится в другой посылке. Уровень срабатывания равен 6.

6. Определение угла по его тангенсу.

   $\forall_{ax}(0 \le \pi/2 - x \ \& \ 0 \le \pi/2 + x \to \operatorname{tg} x = a \leftrightarrow x = \operatorname{arctg} a)$

   Уравнение входит в посылку задачи на исследование, имеющей цель "известно". Выражение x содержит неизвестные, выражение a - не содержит. Так как в геометрических задачах обычно рассматриваются неотрицательные величины углов, то данный прием применяется только в задачах по физике, где отрицательные углы появляются существенно чаще (например, они допустимы при характеризации наклонной плоскости). Для определения угла в геометрических задачах созданы другие приемы, которые будут рассмотрены в соответствующих разделах. Уровень срабатывания данного приема равен 2.

7. Модуль под тангенсом.

   $\forall_a(\operatorname{tg}|a| = \operatorname{tg} a \ \operatorname{sg}(a))$

   Тангенс входит в посылку задачи на исследование, имеющую цель "известно". Выражение a содержит неизвестные. Уровень срабатывания равен 2.

Нормализатор общей стандартизации "нормтангенс"

Нормализатор содержит следующие приемы, дублирующие рассмотренные выше приемы сканирования задачи:

1. Тангенс константы: $0, \pi/6, \pi/4, \pi/3$.
2. Минус под тангенсом.
3. Тангенс целого кратного пи.
4. Формулы приведения.
5. Преобразование дробного слагаемого аргумента к виду суммы дробей.
6. Тангенс арктангенса.
7. Выражение тангенса через длины катетов прямоугольного треугольника.

   $\forall_{ABC}$(прямая$(AC)\perp$прямая(BC) & разныеточки$(A, C) \rightarrow \mathrm{tg}(\angle(BAC)) = l(BC)/l(AC))$

   Первый антецедент обрабатывается идентифицирующим оператором, второй - проверочным. Вместо условия $\neg(A = C)$ в приеме используется вспомогательный терм "разныеточки(A, C)", заголовок которого совпадает с заголовком проверочного оператора, усматривающего различие двух точек. Это избавляет компилятор от необходимости определять типы сравниваемых объектов. В комментарии нормализатора должен входить символ "геометрия".

11.4 Приемы символа "котангенс"

Выражение "котангенс(a)", прорисовываемое формульным редактором в виде $\mathrm{ctg}\, a$, обозначает тангенс вещественного числа a. Большинство приемов символа "котангенс" аналогичны приемам символа "тангенс". Как и в случае символа "косинус", будем в этом разделе приводить приемы практически без комментариев.

Общая стандартизация выражений

1. Котангенс константы.
   (a) $\mathrm{ctg}\, \pi/2 = 0$
   (b) $\mathrm{ctg}\, \pi/6 = \sqrt{3}$
   (c) $\mathrm{ctg}\, \pi/4 = 1$
   (d) $\mathrm{ctg}\, \pi/3 = 1/sqrt3$

   Эти приемы имеют уровень срабатывания 0

   (e) $\mathrm{ctg}\, \pi/8 = \sqrt{2} + 1$
   (f) $\mathrm{ctg}\, \pi/12 = 2 + \sqrt{3}$

   Если прием применяется при упрощении ответа задачи на вычисление, то уровень срабатывания равен 3, иначе он равен 6.

2. Минус под котангенсом.

   $\forall_a(\operatorname{ctg}(-a) = -\operatorname{ctg} a)$

   Уровень срабатывания равен 0.

3. Преобразование дробного слагаемого аргумента к виду суммы дробей.

   $\forall_{abcdefn}(f = (a+b)^n c/d + e \ \& \ \neg(\sin((a+b)^n c/d+e) = 0) \to \operatorname{ctg}((a+b)^n c/d+e) = \operatorname{ctg} f)$

   Аналогично случаю тангенса.

4. Формулы приведения.

   $\forall_{abcde}(e = bc + d \ \& \ 0 < c \to \operatorname{ctg}(a + \pi e/b) = \operatorname{ctg}(a + \pi d/b))$

   $\forall_{abcd}(c = 2a - b \ \& \ 0 \leq c \to \operatorname{ctg}(d + \pi a/b) = -\operatorname{tg}(d + \pi c/2b))$

   $\forall_{abcdef}(a = bc + e \ \& \ f = b - e \to \operatorname{ctg}(d - \pi a/b) = \operatorname{ctg}(d + \pi f/b))$

   $\forall_{abcde}(c = 4e - b \ \& \ 0 \leq c \ \& \ d = b - 2e \to \operatorname{ctg}(a + \pi e/b) = \operatorname{tg}(-a + \pi d/2b))$

   $\forall_{an}(n - \text{целое} \to \operatorname{ctg}(a + \pi n) = \operatorname{ctg} a)$

   Аналогично случаю тангенса.

   $\forall_a(\operatorname{ctg}((a+\pi)/2) = -\operatorname{tg}(a/2))$

   Уровень срабатывания равен 0.

5. Произведение котангенса на синус.

   $\forall_{abcdef}(d = \min(b, c) \ \& \ e = b - d \ \& \ f = c - d \to (\operatorname{ctg} a)^b (\sin a)^c = (\operatorname{ctg} a)^e (\cos a)^d (\sin a)^f)$

   $\forall_{ab}(0 \leq \sin b \to \sqrt{a \operatorname{ctg} b} \sin b = \sqrt{a \sin(2b)/2})$

   Аналогично случаю тангенса.

   $\forall_{abc}(\cos b - \sin b \operatorname{ctg} c = \sin(c - b)/\sin c)$

   Преобразуемая сумма входит в условие задачи на преобразование. Выражение c не содержит неизвестных. Уровень срабатывания равен 1.

6. Произведение тангенса на котангенс.

   $\forall_{ab}((\operatorname{tg} a)^b (\operatorname{ctg} a)^b = 1)$

   Уровень срабатывания равен 0.

   $\forall_{ab}(\operatorname{tg}(a - b) \operatorname{ctg}(b - a) = -1)$

   Уровень срабатывания равен 1.

   $\forall_{amnk}(k = m - n \ \& \ 0 < k \to (\operatorname{tg} a)^m (\operatorname{ctg} a)^n = (\operatorname{tg} a)^k)$

   $\forall_{amnk}(k = m - n \ \& \ 0 < k \to (\operatorname{ctg} a)^m (\operatorname{tg} a)^n = (\operatorname{ctg} a)^k)$

   Переменные m, n идентифицируются с натуральными константами. Антецеденты выделены указателем "программа". Уровень срабатывания равен 0.

   $\forall_a(|\operatorname{ctg} a| \operatorname{tg} a = \operatorname{sg}(\operatorname{tg} a)$

   $\forall_a(|\operatorname{tg} a| \operatorname{ctg} a = \operatorname{sg}(\operatorname{tg} a)$

   Уровень срабатывания равен 1.

7. Сумма тангенса и котангенса.

   $\forall_{ab}(a \operatorname{tg} b + a \operatorname{ctg} b = 2a/\sin(2b))$

   Сумма входит в посылку задачи на исследование, имеющей цель "известно". Уровень срабатывания равен 2.

8. Деление на котангенс.

   $\forall_{bcdg}(d/g(\operatorname{ctg} c)^b = d(\operatorname{tg} c)^b/g)$

   Выражение входит в условие задачи на преобразование. Уровень срабатывания равен 0.

   $\forall_{abc}(a/b \operatorname{ctg} c = a \operatorname{tg} c/b)$

   Прием блокируется, если преобразуется условие задачи на описание, выражение c содержит неизвестные и не существует тригонометрической операции с неизвестными, отличной от $\operatorname{ctg} c$. Уровень срабатывания равен 0.

9. Сигнум котангенса.

   $\forall_a(\operatorname{sg}(\operatorname{ctg} a) = \operatorname{sg}(\operatorname{tg} a))$

   Уровень срабатывания равен 1.

10. Произведение котангенса на синус четырехкратного угла.

    $\forall_a(\operatorname{ctg} a \sin(4a) = 4\cos(2a)(\cos a)^2)$

    Уровень срабатывания равен 3.

11. Сумма с квадратами котангенса и косинуса.

    $\forall_{ab}(a(\cos b)^2(\operatorname{ctg} b)^2 + a(\cos b)^2 = a(\operatorname{ctg} b)^2)$

    Уровень срабатывания равен 3.

12. Редактирование ответа задачи на вычисление.

    (a) Раскрывание скобок для получения произведения тангенса на котангенс.

    $\forall_{abc}((a \operatorname{tg} c + b) \operatorname{ctg} c = a + b \operatorname{ctg} c)$

    Прием применяется в задачах на преобразование, имеющих цель "учетрезультата" либо "известны". Уровень срабатывания равен 3.

    (b) Котангенс половины арксинуса.

    $\forall_a(\neg(a = 0) \to \operatorname{ctg}(\arcsin a/2) = a/(1 - \sqrt{1 - a^2}))$

    Прием применяется в задачах на преобразование, имеющих цель "известны". Для сопровождения по о.д.з. выводятся утверждения $0 \leq 1 - a^2$, $\neg(1 - \sqrt{1 - a^2} = 0)$. Уровень срабатывания равен 2.

    (c) Котангенс в знаменателе.

    $\forall_{abcn}(a/b(\operatorname{ctg} c)^n = a(\operatorname{tg} c)^n/b)$

    Дробь находится в условии задачи на преобразование, имеющей цель "известны". Уровень срабатывания равен 3.

    (d) Переход к котангенсу от частного косинуса и синуса.

    $\forall_{abcd}(a(\cos b)^d/c(\sin b)^d = a(\operatorname{ctg} b)^d/c)$

    Прием применяется в задачах на преобразование, имеющих цель "учетрезультата". Коэффициенты a, c не содержат выражений $\sin b, \cos b$. Уровень срабатывания равен 3.

Выражение котангенса через тангенс

$\forall_a(\neg(\sin a = 0) \to \operatorname{ctg} a = 1/\operatorname{tg} a)$

На этой теореме созданы два приема. Первый из них применяется в условиях задач на преобразование. Необходимо, чтобы выражение $\operatorname{tg} a$ уже имелось в условии задачи. Если преобразуемый котангенс (быть может, возведенный в степень) умножается на степень тангенса, то уровень срабатывания равен 1, иначе он равен 5.

Второй прием применяется в условиях задач на описание, не используемых для сопровождения по о.д.з. Условие должно иметь вхождение выражения $\operatorname{tg} a$, причем либо в этом условии отсутствуют неизвестные тригонометрические операции, отличные от $\operatorname{tg} a, \operatorname{ctg} a$, либо рассматриваемый котангенс расположен в основании степени, имеющей неизвестный показатель, и имеется степень тангенса a с неизвестным показателем. Уровень срабатывания равен 1.

Усмотрение котангенса суммы

В рассматривавшихся задачах понадобился лишь единственный прием такого типа:

$$\forall_{abcde}(\neg(\operatorname{tg} a{+}1 = 0) \,\&\, \neg(\cos a = 0) \to d(b{-}b\operatorname{tg} a)^c/e(\operatorname{tg} a{+}1)^c = db^c(\operatorname{ctg}(a{+}\pi/4))^c/e)$$

Дробь расположена в условии задачи на преобразование. Вне нее не должно быть вхождений тригонометрических операций с аргументом a. Уровень срабатывания равен 3.

Выражение котангенса через синус и косинус

$\forall_a(\operatorname{ctg} a = \cos a/\sin a)$

На этой теореме созданы два приема. Первый из них применяется к котангенсу, входящему в константное условие задачи на преобразование, имеющей цель "упростить". Условие должно иметь тригонометрический аргумент, отличный от a. Уровень срабатывания равен 4. Второй прием применяется при упрощении подынтегрального выражения. Переменная интегрирования должна входить в a. Если котангенс расположен под логарифмом, прием блокируется. Уровень срабатывания тоже равен 4.

Равенство котангенса нулю

$\forall_a(\operatorname{ctg} a = 0 \leftrightarrow \cos a = 0)$

Уровень срабатывания равен 2.

Котангенс выражения, содержащего обратные тригонометрические функции

1. Котангенс арксинуса.

   $\forall_a(\neg(a = 0) \,\&\, 0 \le 1 - a \,\&\, 0 \le 1 + a \to \operatorname{ctg} \arcsin a = \sqrt{1 - a^2}/a)$

   Если выражение a константное, уровень срабатывания равен 0, иначе он равен 1. Если преобразуется подвыражение посылки задачи на доказательство или на исследование, либо условия задачи на преобразование, имеющей цель "известны", то сопровождение по о.д.з. не требуется, и проверка антецедентов отбрасывается:

$\forall_a(\operatorname{ctg}\arcsin a = \sqrt{1-a^2}/a)$

Уровень срабатывания здесь равен 0.

2. Котангенс арккосинуса.

   $\forall_a(0 < 1+a \ \& \ 0 < 1-a \to \operatorname{ctg}\arccos a = a/\sqrt{1-a^2})$

   $\forall_a(\operatorname{ctg}\arccos a = a/\sqrt{1-a^2})$

   Аналогично предыдущему, но первый прием имеет только уровень срабатывания 1.

3. Котангенс арктангенса.

   $\forall_a(\neg(a=0) \to \operatorname{ctg}\operatorname{arctg} a = 1/a)$

   $\forall_a(\operatorname{ctg}\operatorname{arctg} a = 1/a)$

   Второй прием применяется в посылках задач на доказательство или на исследование, а также в условиях задач на преобразование, имеющих цель "известны". Уровни срабатывания обоих приемов равны 0.

4. Котангенс половины арксинуса.

   $\forall_a(\neg(a=0) \ \& \ 0 \le 1+a \ \& \ 0 \le 1-a \to \operatorname{ctg}(\arcsin a/2) = a/(1-\sqrt{1-a^2}))$

   Уровень срабатывания равен 1.

5. Котангенс половины арккосинуса.

   $\forall_a(0 < 1-a \ \& \ 0 \le 1+a \to \operatorname{ctg}(\arccos a/2) = \sqrt{(1+a)/(1-a)})$

   Уровень срабатывания равен 2.

6. Котангенс половины арктангенса.

   $\forall_a(\neg(a=0) \to \operatorname{ctg}(\operatorname{arctg} a/2) = a/(\sqrt{a^2+1}-1))$

   Уровень срабатывания равен 2.

7. Использование формулы котангенса суммы.

   $\forall_{ab}(\neg(\cos b = 0) \ \& \ \neg(\cos a = 0) \ \& \ \neg(\sin(a+b) = 0) \to \operatorname{ctg}(a+b) = (1 - \operatorname{tg} a \operatorname{tg} b)/(\operatorname{tg} a + \operatorname{tg} b))$

   Котангенс входит в условие задачи на преобразование либо на описание, причем в последнем случае - при редактировании ответа. Слагаемое b имеет сомножителем своего числителя обратную тригонометрическую функцию. Уровень срабатывания равен 2.

Уравнения

1. Простейшее уравнение для котангенса.

   $\forall_{ab}(\operatorname{ctg} a = b \leftrightarrow \exists_n(n - \text{целое} \ \& \ a = -\operatorname{arctg} b + \pi/2 + \pi n))$

   Уравнение является условием задачи на описание. Выражение a содержит неизвестные, b - не содержит. Оператор "фильтрсерии" ложен. Уровень срабатывания равен 2.

$\forall_{ab}(0 \leq a \ \& \ 0 \leq \pi - a \rightarrow \operatorname{ctg} a = b \leftrightarrow a = \pi/2 - \operatorname{arctg} b)$

Уравнение является условием задачи на описание, имеющей цель "стандравно". Уровень срабатывания равен 0.

2. Замена отрицания равенства нулю котангенса на отрицание равенства нулю косинуса.

   $\forall_a(\neg(\sin a = 0) \rightarrow \neg(\operatorname{ctg} a = 0) \leftrightarrow \neg(\cos a = 0))$

   Замена выполняется даже в тех случаях, когда преобразуемый терм задачи используется для сопровождения по о.д.з. Уровень срабатывания равен 1.

3. Выражение котангенса через синус и косинус.

   $\forall_a(\operatorname{ctg} a = \cos a / \sin a)$

   Котангенс входит в условие задачи на описание, имеющей более одной неизвестной. Выражение a содержит неизвестные. Существует уравнение, содержащее $\sin a$ либо $\cos a$. Уровень срабатывания равен 2.

4. Разность тангенса и котангенса.

   $\forall_{ab}(\neg(\sin b = 0) \ \& \ \neg(\cos b = 0) \rightarrow a \operatorname{tg} b - a \operatorname{ctg} b = -2a \operatorname{ctg}(2b))$

   Разность входит в условие задачи на описание. Уровень срабатывания равен 1.

Неравенства

1. Решение простейших нестрогих неравенств с котангенсом.

   $\forall_{ab}(\neg(\sin b = 0) \rightarrow a \leq \operatorname{ctg} b \leftrightarrow \exists_n(n - \text{целое} \ \& \ \pi n < b \ \& \ b \leq \pi n + \operatorname{arcctg} a))$

   $\forall_{ab}(\neg(\sin b = 0) \rightarrow \operatorname{ctg} b \leq a \leftrightarrow \exists_n(n - \text{целое} \ \& \ \pi n + \operatorname{arcctg} a \leq b \ \& \ b < \pi n + \pi))$

   Неравенство является условием задачи на описание, не имеющей уравнений и не имеющей цели "стандравно". Это условие не используется для сопровождения по о.д.з. Выражение a содержит неизвестные, b - не содержит. Оператор "фильтрпромежутков" ложен. Уровни срабатывания равны 2 и 6.

2. Решение простейших строгих неравенств с котангенсом.

   $\forall_{ab}(\neg(\sin b = 0) \rightarrow a < \operatorname{ctg} b \leftrightarrow \exists_n(n - \text{целое} \ \& \ \pi n < b \ \& \ b < \pi n + \operatorname{arcctg} a))$

   $\forall_{ab}(\neg(\sin b = 0) \rightarrow \operatorname{ctg} b < a \leftrightarrow \exists_n(n - \text{целое} \ \& \ \pi n + \operatorname{arcctg} a < b \ \& \ b < \pi n + \pi))$

Задачи на исследование, имеющие цель "известно"

1. Сокращение на косинус.

   $\forall_{abc}(\neg(\cos a = 0) \rightarrow b \operatorname{ctg} a = c \cos a \leftrightarrow b = c \sin a)$

   Равенство входит в посылку задачи на доказательство либо на исследование. Уровень срабатывания равен 2.

2. Использование равенства с котангенсом для преобразования соотношения пропорциональности синуса и косинуса.

   $\forall_{abcdx}(x = \operatorname{ctg} p \rightarrow a \sin p/b + c \cos p/d = 0 \leftrightarrow a/b + cx/d = 0)$

   $\forall_{abcdx}(x = \operatorname{ctg} p \rightarrow a \sin p/b = c \cos p/d \leftrightarrow a/b = cx/d)$

Равенство с синусом и косинусом входит в посылку задачи на исследование, имеющей цель "известно". Антецедент идентифицируется с другой посылкой задачи. Выражения a, b, c, d не содержат неизвестных, p - содержит. Переменная x идентифицируется с выражением, все неизвестные которого суть неизвестные внешней задачи на описание. Оно содержит хотя бы одну неизвестную, но не имеет неизвестных тригонометрических операций. Уровень срабатывания равен 2.

Нормализатор общей стандартизации "нормкотангенс"

1. Котангенс константы: $\pi/2$, $\pi/6$, $\pi/4$, $\pi/3$.
2. Минус под котангенсом.
3. Формулы приведения.
4. Преобразование дробного слагаемого аргумента к виду суммы дробей.

11.5 Приемы символа "секанс"

Выражение "секанс(a)", прорисовываемое формульным редактором в виде $\sec a$, обозначает секанс вещественного числа a. Операции "секанс", "косеканс" обычно сразу же выражаются через косинус и синус. Поэтому приемов, связанных с этими операциями, крайне мало.

Секанс константы

1. $\sec 0 = 1$
2. $\sec \pi/6 = 2/\sqrt{3}$
3. $\sec \pi/4 = \sqrt{2}$
4. $\sec \pi/3 = 2$

Перечисленные приемы имеют уровень срабатывания 0.

5. $\sec \pi/8 = 2/\sqrt{2+\sqrt{2}}$
6. $\sec \pi/12 = 2\sqrt{2}/(1+\sqrt{3})$

Последние два приема срабатывают на уровне 6. Они рассчитаны на те исключительные ситуации, когда секанс по каким-либо причинам не выражен через косинус.

Минус под секансом

$\forall_a(\sec(-a) = \sec a)$

Уровень срабатывания равен 0.

Выражение секанса через косинус

$\forall_a(\sec a = 1/\cos a)$

Рассматриваемое вхождение секанса не расположено под интегралом. Уровень срабатывания равен 1.

Нормализатор общей стандартизации "нормсеканс"

1. Секанс константы: 0, $\pi/6$, $\pi/4$, $\pi/3$.
2. Минус под секансом.

11.6 Приемы символа "косеканс"

Выражение "косеканс(a)", прорисовываемое формульным редактором в виде $\operatorname{cosec} a$, обозначает косеканс вещественного числа a.

Косеканс константы

1. $\operatorname{cosec}\pi/2 = 1$
2. $\operatorname{cosec}\pi/6 = 2$
3. $\operatorname{cosec}\pi/4 = \sqrt{2}$
4. $\operatorname{cosec}\pi/3 = 2/\sqrt{3}$

   Перечисленные приемы имеют уровень срабатывания 0.

5. $\operatorname{cosec}\pi/8 = 2/\sqrt{2-\sqrt{2}}$
6. $\operatorname{cosec}\pi/12 = 2\sqrt{2}/(\sqrt{3}-1)$

   Последние два приема срабатывают на уровне 6.

Минус под косекансом

$\forall_a(\operatorname{cosec}(-a) = -\operatorname{cosec} a)$

Уровень срабатывания равен 0.

Выражение косеканса через синус

$\forall_a(\operatorname{cosec} a = 1/\sin a)$

Уровень срабатывания равен 1.

Нормализатор общей стандартизации "нормкосеканс"

1. Косеканс константы: $\pi/2$, $\pi/6$, $\pi/4$, $\pi/3$.
2. Минус под косекансом.

11.7 Приемы символа "арксинус"

Выражение "арксинус(a)", прорисовываемое формульным редактором в виде $\arcsin a$, обозначает арксинус вещественного числа a. Справочники "тип", "арность", "одз", "типданных" характеризуют простейшие свойства символа "арксинус".

Общая стандартизация выражений

1. Арксинус константы.
   (a) $\arcsin 0 = 0$
   (b) $\arcsin 1 = \pi/2$
   (c) $\arcsin 1/2 = \pi/6$
   (d) $\arcsin 1/\sqrt{2} = \pi/4$
   (e) $\arcsin \sqrt{3}/2 = \pi/3$
   (f) $\arcsin(\sqrt{5}-1)/4 = \pi/10$
   (g) $\arcsin(\sqrt{5}+1)/4 = 3\pi/10$
   (h) $\arcsin(\sqrt{3}-1)/2\sqrt{2}) = \pi/12$

   Уровень срабатывания перечисленных приемов равен 0.

2. Минус под арксинусом.

   $\forall_a(\arcsin(-a) = -\arcsin a)$

   Уровень срабатывания равен 0.

3. Арксинус синуса.

   $\forall_a(0 \leq 2a + \pi \ \& \ 0 \leq \pi - 2a \rightarrow \arcsin\sin a = a)$

   Антецеденты обрабатываются проверочным оператором. Уровень срабатывания равен 1.

   $\forall_{ak}(k = [(2a+\pi)/2\pi] \rightarrow \arcsin\sin a = (-1)^k a - \pi(-1)^k k)$

   Выражение входит в условие задачи на преобразование. Антецедент выделен указателем "идентификатор", его правая часть обрабатывается нормализаторами общей стандартизации. Уровень срабатывания равен 2.

4. Арксинус косинуса.

   $\forall_a(0 \leq a \ \& \ 0 \leq \pi - a \rightarrow \arcsin\cos a = \pi/2 - a)$

   Антецеденты обрабатываются проверочными операторами. Уровень срабатывания равен 1.

5. Арксинус модуля.

   $\forall_a(\arcsin|a| = |\arcsin a|)$

   Уровень срабатывания равен 1.

Общая стандартизация утверждений

1. Знак арксинуса.

   $\forall_a(0 \leq \arcsin a \leftrightarrow 0 \leq a)$

   $\forall_a(\arcsin a \leq 0 \leftrightarrow a \leq 0)$

   Допускаются строгие неравенства. Уровень срабатывания равен 0.

2. Усмотрение простейшего неравенства с арксинусом.

   $\forall_{abcd}(0 \leq 2a|d| - |b|\pi \rightarrow 0 \leq a + (b \arcsin c)/d)$

   Прием имеет заголовок "второйтерм". Выражения a, b, d константные. Допускаются вырожденные единичные значения b, d. Антецедент обрабатывается проверочным оператором. Уровень срабатывания равен 1.

3. Исключение арксинуса в простейших неравенствах.

   $\forall_{ab}(0 \leq \pi - 2a \ \& \ 0 \leq \pi + 2a \rightarrow \arcsin b < a \leftrightarrow b < \sin a)$

   $\forall_{ab}(0 \leq \pi - 2a \ \& \ 0 \leq \pi + 2a \rightarrow \arcsin b \leq a \leftrightarrow b \leq \sin a)$

   Выражение a константное, b - неконстантное. Антецеденты обрабатываются проверочными операторами. Допускается перестановка частей неравенства. Уровень срабатывания равен 2.

   $\forall_{ab}(\arcsin a \leq \arcsin b \leftrightarrow a \leq b)$

   Допускается строгое неравенство. Уровень срабатывания равен 0.

   $\forall_{ab}(0 \leq \arcsin a - \arcsin b \leftrightarrow b \leq a$

   Допускается строгое неравенство. Уровень срабатывания равен 1.

4. Равенство арксинусов.

   $\forall_{ab}(\arcsin a = \arcsin b \leftrightarrow a = b)$

   $\forall_{ab}(\arcsin a - \arcsin b = 0 \leftrightarrow a - b = 0)$

   Уровень срабатывания первого приема равен 1, второго - 0.

5. Сумма арксинусов равна пи.

   $\forall_{ab}(\arcsin a = \pi - \arcsin b \leftrightarrow a = 1 \ \& \ b = 1)$

   Уровень срабатывания равен 1.

Разность арксинусов

$\forall_{abc}(0 \leq a \ \& \ 0 \leq 1 - a \ \& \ 0 \leq b \ \& \ 0 \leq 1 - b \rightarrow c \arcsin a - c \arcsin b = c \arcsin(a\sqrt{1-b^2} - b\sqrt{1-a^2}))$

Либо разность входит в условие задачи на преобразование, не имеющей цели "редакция", и тогда уровень срсбатывания равен 2, либо она входит в уравнение, являющееся условием задачи на описание, имеющей цель "редакция", выражения a, b константные, а уровень срабатывания равен 0. В обоих случаях выражение под заменяющим арксинусом упрощается с помощью вспомогательной задачи на преобразование.

Сумма арксинусов

$\forall_{abc}(0 \leq a \;\&\; 0 \leq 1-a \;\&\; 0 \leq b \;\&\; 0 \leq 1-b \rightarrow c \arcsin a + c \arcsin b = c \arccos(\sqrt{1-a^2}\sqrt{1-b^2} - ab))$

Аналогично предыдущему.

Сумма арксинуса и арккосинуса

$\forall_{abc}(0 \leq a \;\&\; 0 \leq 1-a \;\&\; 0 \leq b \;\&\; 0 \leq 1-b \rightarrow c \arcsin a + c \arccos b = c \arccos(b\sqrt{1-a^2} - a\sqrt{1-b^2}))$

Аналогично предыдущему.

$\forall_a(0 \leq a+1 \;\&\; 0 \leq 1-a \rightarrow \arcsin a + \arccos a = \pi/2)$

Уровень срабатывания равен 0.

Разность арксинуса и арккосинуса

$\forall_{abc}(0 \leq a \;\&\; 0 \leq 1-a \;\&\; 0 \leq b \;\&\; 0 \leq 1-b \rightarrow c \arcsin a - c \arccos b = c \arcsin(ab - \sqrt{1-a^2}\sqrt{1-b^2}))$

Аналогично предыдущему.

Удвоенный арксинус

$\forall_a(0 \leq a \;\&\; 0 \leq 1-a \rightarrow 2 \arcsin a = \arccos(1-2a^2))$

Либо удвоенный арксинус входит в условие задачи на преобразование, и тогда уровень срабатывания равен 2, либо он входит в уравнение с другими вхождениями обратных тригонометрических функций, являющееся условием задачи на описание, выражение a константное, а уровень срабатывания равен 0. Рассматриваемое произведение не является операндом тригонометрической операции. Если выражение a неконстантное, то его числитель должен содержать неконстантный радикал.

Простейшее уравнение с арксинусом

$\forall_{ab}(\arcsin a = b \leftrightarrow a = \sin b \;\&\; 0 \leq \pi - 2b \;\&\; 0 \leq \pi + 2b)$

Уравнение является условием задачи на описание. Выражение a содержит неизвестные, b - не содержит. Уровень срабатывания равен 1.

Простейшие неравенства с арксинусом

$\forall_{ax}(0 \leq 1+x \;\&\; 0 \leq 1-x \rightarrow \arcsin x < a \leftrightarrow \pi/2 < a \vee -\pi/2 < a \;\&\; a \leq \pi/2 \;\&\; x < \sin a)$

$\forall_{ax}(0 \leq 1+x \;\&\; 0 \leq 1-x \rightarrow a < \arcsin x \leftrightarrow a < -\pi/2 \vee -\pi/2 \leq a \;\&\; a < \pi/2 \;\&\; \sin a < x)$

Неравенство входит в условие задачи на описание. Выражение x содержит неизвестные, выражение a - не содержит. Разрешаются случаи нестрогих неравенств. Антецеденты обрабатываются проверочными операторами. Уровень срабатывания равен 3.

Вывод следствия из уравнения с арксинусом

$\forall_{abc}(\arcsin a + b = c \to a - \sin(c - b) = 0)$

Прием имеет заголовок "выводусловия". Антецедент идентифицируется с условием задачи на описание. Выражения a, b содержат неизвестные. Первое из них имеет вхождение синуса либо косинуса. Выражение c неизвестных не содержит. Уровень срабатывания равен 8.

Нормализатор общей стандартизации "нормарксинус"

1. Арксинус константы: $0, 1, 1/2, 1/\sqrt{2}, \sqrt{3}/2, (\sqrt{5} \pm 1)/4, (\sqrt{3} - 1)/2\sqrt{2}$
2. Минус под арксинусом.
3. Арксинус синуса.

11.8 Приемы символа "арккосинус"

Выражение "арккосинус(a)", прорисовываемое формульным редактором в виде $\arccos a$, обозначает арккосинус вещественного числа a. Приемы этого раздела аналогичны приемам предыдущего.

Общая стандартизация выражений

1. Арккосинус константы
   - (a) $\arccos 0 = \pi/2$
   - (b) $\arccos 1 = 0$
   - (c) $\arccos 1/2 = \pi/3$, $\arccos 0.5 = \pi/3$
   - (d) $\arccos 1/\sqrt{2} = \pi/4$
   - (e) $\arccos \sqrt{3}/2 = \pi/6$
   - (f) $\arccos(\sqrt{5} - 1)/4 = 2\pi/5$
   - (g) $\arccos(\sqrt{5} + 1)/4 = \pi/5$
   - (h) $\arccos \sqrt{1 + \sqrt{2}}/2^{3/4} = \pi/8$
   - (i) $\arccos(1 + \sqrt{3})/2\sqrt{2} = \pi/12$
   - (j) $\arccos(\sqrt{3} - 1)/2\sqrt{2} = 5\pi/12$
   - (k) Попытка устранения двойного радикала под арккосинусом путем перехода к арксинусу.
     $\forall_{abcde}(e = \sqrt{d^2 - a - b\sqrt{c}} \to \arccos \sqrt{a + b\sqrt{c}}/d = \arcsin e/d)$
     Выражения a, b, c, d идентифицируются с десятичными константами. Антецедент выделен указателем "идентификатор". Его правая часть обрабатывается вспомогательной задачей на упрощение. Проверяется, что результат e не имеет двойных радикалов. Уровень срабатывания равен 3.
   - (l) $\arccos(\sqrt[4]{5}\sqrt{1 + \sqrt{5}}/2\sqrt{2} = \pi/10$

2. Минус под арккосинусом.

   $\forall_a(\arccos(-a) = \pi - \arccos a)$

   Если выражение входит в условие задачи на описание, то уровень срабатывания равен 2, иначе он равен 0.

3. Арккосинус косинуса.

   $\forall_a(0 \leq a \ \& \ 0 \leq \pi - a \rightarrow \arccos\cos a = a)$

   Уровень срабатывания равен 1.

   $\forall_{ab}(b = [a/\pi] \rightarrow \arccos\cos a = (-1)^b a + 2\pi(-1)^b[-b/2])$

   Арккосинус входит в условие задачи на преобразование, не имеющей цели "редакция". Антецедент выделен указателем "идентификатор", его правая часть обрабатывается нормализаторами общей стандартизации. Уровень срабатывания равен 2.

4. Арккосинус синуса.

   $\forall_a(0 \leq \pi + 2a \ \& \ 0 \leq \pi - 2a \rightarrow \arccos\sin a = \pi/2 - a)$

   Уровень срабатывания равен 1.

   $\forall_a(0 \leq a \ \& \ 0 \leq \pi - a \rightarrow \arccos\sin a = |\pi/2 - a|)$

   Уровень срабатывания равен 2.

5. Вычитание арккосинуса из пи.

   $\forall_{abcd}(\pi - \arccos((a-b)c/d) = \arccos((b-a)c/d))$

   Разность входит в условие задачи на преобразование, имеющей цель "известны", т.е. имеет место завершающее редактирование ответа задачи на вычисление. Уровень срабатывания равен 0.

6. Половина арккосинуса.

   $\forall_{ab}((\arccos a + b)/2 = \arccos(\sqrt{(a+1)/2}) + b/2)$

   $\forall_{ab}((-\arccos a + b)/2 = -\arccos(\sqrt{(a+1)/2}) + b/2)$

   Выражение a константное. Дробь входит в условие задачи на преобразование, имеющей цель "известны". Уровень срабатывания равен 2.

Общая стандартизация утверждений

1. Усмотрение простейшего неравенства с арккосинусом. Здесь созданы два приема, имеющие заголовок "второйтерм".

   $\forall_{ab}(0 \leq a \ \& \ 0 \leq c \rightarrow 0 \leq a \arccos b + c)$

   Выражения a, b константные, антецеденты обрабатываются проверочными операторами. Уровень срабатывания равен 0.

   $\forall_{abcd}(0 \leq a|d| - |b|\pi \rightarrow 0 \leq a + (b \arccos c)/d)$

   Выражения a, b, d константные. Антецедент обрабатывается проверочным оператором. Уровень срабатывания равен 1.

2. Равенство арккосинуса нулю.

   $\forall_a(\arccos a = 0 \leftrightarrow a = 1)$

   Уровень срабатывания равен 0.

3. Равенство арккосинусов.

   $\forall_{ab}(\arccos a = \arccos b \leftrightarrow a = b)$

   $\forall ab(\arccos a - \arccos b = 0 \leftrightarrow a = b)$

   Уровень срабатывания равен 0.

4. Сумма арккосинусов равна 0.

   $\forall_{ab}(-\arccos a = \arccos b \leftrightarrow a = 1 \mathbin{\&} b = 1)$

   Уровень срабатывания равен 0.

5. Исключение арккосинуса в простейших неравенствах.

   $\forall_{ab}(0 < \arccos a - \arccos b \leftrightarrow a < b)$

   $\forall_{ab}(0 \leq \arccos a - \arccos b \leftrightarrow a \leq b)$

   Уровень срабатывания равен 1.

   $\forall_{ab}(0 < a - \pi \rightarrow \arccos b < a)$

   $\forall_{ab}(0 \leq a - \pi \rightarrow \arccos b \leq a)$

   Антецедент обрабатывается проверочным оператором, уровень срабатывания равен 2.

   $\forall_a(\pi/2 < \arccos a \leftrightarrow a < 0)$

   $\forall_a(\pi/2 \leq \arccos a \leftrightarrow a \leq 0)$

   Уровень срабатывания равен 2.

   $\forall_{ab}(0 \leq \pi - a \mathbin{\&} 0 \leq a \rightarrow \arccos b \leq a \leftrightarrow \cos a \leq b)$

   $\forall_{ab}(0 \leq \pi - a \mathbin{\&} 0 \leq a \rightarrow \arccos b < a \leftrightarrow \cos a < b)$

   Выражение b неконстантное, a - константное. Антецеденты обрабатываются проверочными операторами. Допускается перестановка частей неравенства. Уровень срабатывания равен 2.

Сумма арккосинусов

$\forall_{abc}(0 \leq a \mathbin{\&} 0 \leq 1 - a \mathbin{\&} 0 \leq c \mathbin{\&} 0 \leq 1 - c \rightarrow b \arccos a + b \arccos c = b \arccos(ac - \sqrt{1 - a^2}\sqrt{1 - c^2}))$

Аналогично случаю арксинусов.

Разность арккосинусов

$\forall_{abc}(0 \leq a \mathbin{\&} 0 \leq 1 - a \mathbin{\&} 0 \leq c \mathbin{\&} 0 \leq 1 - c \rightarrow b \arccos a - b \arccos c = b \arcsin(c\sqrt{1 - a^2} - a\sqrt{1 - c^2}))$

Аналогично случаю арксинусов.

$\forall_{abcde}(0 \leq a \ \& \ 0 \leq 1-a \ \& \ 0 \leq c \ \& \ 0 \leq 1-c \ \& \ d = b\arccos a - b\arccos c \ \& \ e = d - b\arccos c \rightarrow b\arccos a - 2b\arccos c = e)$

Здесь происходит двукратное вычитание арккосинуса. Выражения a, c константные. Первые четыре антецедента обрабатываются проверочными операторами. Последние два антецедента выделены указателем "идентификатор". Их правые части упрощаются с помощью вспомогательных задач на преобразование. Проверяется, что выражения d, e не имеют заголовка "плюс". Уровень срабатывания равен 0.

Простейшее уравнение с арккосинусом

$\forall_{ab}(\arccos a = b \leftrightarrow a = \cos b \ \& \ 0 \leq b \ \& \ 0 \leq \pi - b)$

Уравнение является условием задачи на описание. Выражение a содержит неизвестные, b - не содержит. Уровень срабатывания равен 1.

Простейшие неравенства с арккосинусом

$\forall_{ax}(0 \leq 1+x \ \& \ 0 \leq 1-x \rightarrow \arccos x < a \leftrightarrow \pi < a \ \vee \ 0 < a \ \& \ 0 \leq \pi \ \& \ \cos a < x)$

$\forall_{ax}(0 \leq 1+x \ \& \ 0 \leq 1-x \rightarrow a < \arccos x \leftrightarrow a < 0 \ \vee \ 0 \leq a \ \& \ a < \pi \ \& \ x < \cos a)$

Неравенство входит в условие задачи на описание. Выражение x содержит неизвестные, a не содержит. Антецеденты обрабатываются проверочными операторами. Допускается случай нестрогого неравенства. Уровень срабатывания равен 3.

Вывод следствия из уравнения с арккосинусом

$\forall_{abc}(\arccos a + b = c \rightarrow a - \cos(c-b) = 0)$

Аналогично случаю арксинуса.

Выражение арккосинуса через арксинус

$\forall_{a}(0 \leq 1+a \ \& \ 0 \leq 1-a \rightarrow \arccos a = \pi/2 - \arcsin a)$

Арккосинус входит в условие задачи на описание, уже имеющее вхождение выражения $\arcsin a$. Выражение a содержит неизвестные. Уровень срабатывания приема равен 2.

Нормализатор общей стандартизации "нормарккосинус"

1. Арккосинус константы: $0, 1, 1/2, 1/\sqrt{2}, \sqrt{3}/2, (\sqrt{5} \pm 1)/4, \sqrt[4]{5}\sqrt{1+\sqrt{5}}/2\sqrt{2}, (\sqrt{3}+1)/2\sqrt{2}, (\sqrt{2}+1)/2^{3/4}$.
2. Минус под арккосинусом.

11.9 Приемы символа "арктангенс"

Выражение "арктангенс(a)", прорисовываемое формульным редактором в виде $\operatorname{arctg} a$, обозначает арктангенс вещественного числа a. Справочники "тип", "арность", "одз", "типданных" характеризуют простейшие свойства символа "арктангенс".

Общая стандартизация выражений

1. Арктангенс константы.

   (a) $\operatorname{arctg} 0 = 0$
   (b) $\operatorname{arctg} 1 = \pi/4$
   (c) $\operatorname{arctg}(1/\sqrt{3}) = \pi/6$
   (d) $\operatorname{arctg}(\sqrt{3}) = \pi/3$
   (e) $\operatorname{arctg}(2 - \sqrt{3}) = \pi/12$
   (f) $\operatorname{arctg}(\sqrt{2} - 1) = \pi/8$
   (g) $\operatorname{arctg}(\sqrt{3} + 2) = 5\pi/12$
   (h) $\operatorname{arctg}(\sqrt{2} + 1) = 3\pi/8$
   (i) $\operatorname{arctg}(1 - \sqrt{2}) = -\pi/8$
   (j) $\operatorname{arctg}(\sqrt{3} - 2) = -\pi/12$

   Уровни срабатывания равны 0.

2. Минус под арктангенсом.

   $\forall_a(\operatorname{arctg}(-a) = -\operatorname{arctg} a)$

   Уровень срабатывания равен 0.

3. Арктангенс тангенса.

   $\forall_a(0 < \pi - 2a \ \& \ 0 < 2a + \pi \to \operatorname{arctg} \operatorname{tg} a = a)$

   Антецеденты обрабатываются проверочными операторами. Введен сравнительно сильный ограничитель трудоемкости. Уровень срабатывания равен 1.

   $\forall_a(\operatorname{arctg} \operatorname{tg} a = a - \pi[(2a + \pi)/2\pi])$

   Прием применяется в условиях задач на преобразование, не имеющих цели "нормИнтеграл". Уровень срабатывания равен 2. На той же самой теореме создана еще одна версия приема, применяемая в условиях задач на описание при неизвестном a. Ее уровень срабатывания равен 0.

   $\forall_a(\operatorname{arctg} \operatorname{tg} a = a)$

   Прием применяется для предварительных преобразований подынтегрального выражения при формальном интегрировании. Уровень срабатывания равен 2.

4. Арктангенс котангенса.

   $\forall_a(0 \leq a \ \& \ 0 \leq \pi - a \to \operatorname{arctg} \operatorname{ctg} a = \pi/2 - a)$

   $\forall_a(0 \leq a \ \& \ 0 \leq \pi - a \to \operatorname{arctg}(1/\operatorname{tg} a) = \pi/2 - a)$

   Аналогично первом приему для арктангенса тангенса, но уровень срабатывания равен 0.

   $\forall_a(\operatorname{arctg} \operatorname{ctg} a = \pi/2 - a - \pi[(\pi - a)/\pi])$

   Прием применяется в условиях задач на преобразование, не имеющих цели "нормИнтеграл". Уровень срабатывания равен 2.

5. Модуль под арктангенсом.

   $\forall_{abcd}(\operatorname{arctg}(|a|b)c/|a|d = \operatorname{arctg}(ab)c/ad)$

   Допускаются вырожденные единичные значения b, c, d. Уровень срабатывания равен 2.

6. Изменение знака разности под арктангенсом.

   $\forall_{abc}(c = a - b \rightarrow \operatorname{arctg}(b - a) = -\operatorname{arctg} c)$

   Антецедент выделен указателем "идентификатор". Его правая часть обрабатывается нормализаторами "нормплюс" и "стандупорядочение". Результат c должен лексикографически предшествовать исходной разности $b - a$. Если последняя имела слагаемое вида $m\pi/n$ для натуральных m, n, то прием блокируется. Он также блокируется при завершающем редактировании ответа задачи на преобразование. Уровень срабатывания равен 1.

Общая стандартизация утверждений

1. Равенство арктангенсов.

   $\forall_{ab}(\operatorname{arctg} a = \operatorname{arctg} b \leftrightarrow a = b)$

   Уровень срабатывания равен 0.

2. Усмотрение простейших неравенств с арктангенсом. Приводимые ниже приемы имеют заголовок "второйтерм":

   $\forall_{a}(\operatorname{arctg} a \leq \pi/2)$

   $\forall_{a}(-\pi/2 \leq \operatorname{arctg} a)$

   $\forall_{a}(\neg(0 < \operatorname{arctg} a - \pi/2))$

   В последнем приеме допускается случай нестрогого неравенства. Уровень срабатывания равен 1.

   $\forall_{abc}(0 \leq 2c - \pi|b| \rightarrow 0 \leq b \operatorname{arctg} a + c)$

   $\forall_{abc}(0 < 2c - \pi|b| \rightarrow 0 < b \operatorname{arctg} a + c)$

   Выражения b, c константные. Антецедент обрабатывается проверочным оператором. Уровень срабатывания равен 2.

3. Исключение арктангенса в простейших неравенствах.

   $\forall_{a}(\operatorname{arctg} a \leq 0 \leftrightarrow a \leq 0)$

   Допускается случай строгого неравенства. Уровень срабатывания равен 2.

   $\forall_{ab}(0 \leq \operatorname{arctg} a - \operatorname{arctg} b \leftrightarrow 0 \leq a - b)$

   $\forall_{ab}(0 \leq \operatorname{arctg} a + \operatorname{arctg} b \leftrightarrow 0 \leq a + b)$

   Допускаются перестановка частей неравенства и переход к строгому неравенству. Уровень срабатывания равен 0.

Удвоенный арктангенс

$\forall_{abc}(0 < a \ \& \ b = 1 - a^2 \ \& \ 0 < b \to 2c \operatorname{arctg} a = c \operatorname{arctg}(2a/b))$

Либо произведение входит в содержащее более одной обратной тригонометрической операции уравнение - условие задачи на описание, выражение a константное и уровень срабатывания равен 0, либо произведение входит в условие задачи на преобразование, а уровень срабатывания равен 1. В обоих случаях задача должна иметь цель "редакция", причем произведение не должно являться операндом тригонометрической операции. Второй антецедент выделен указателем "идентификатор", первый и третий - обрабатываются проверочными операторами.

$\forall_{abc}(0 < a \ \& \ b = 1 - a^2 \ \& \ b < 0 \to 2c \operatorname{arctg} a = \pi c + c \operatorname{arctg}(2a/b))$

Аналогично предыдущему, но уровни срабатывания для задач на описание и преобразование равны 3.

Сумма арктангенсов

$\forall_{abcde}(0 < a \ \& \ 0 < b \ \& \ c = 1 - ab \ \& \ 0 < c \ \& \ d = a + b \to e \operatorname{arctg} a + e \operatorname{arctg} b = e \operatorname{arctg}(d/c))$

$\forall_{abcde}(0 < a \ \& \ 0 < b \ \& \ c = 1 - ab \ \& \ c < 0 \ \& \ d = a + b \to e \operatorname{arctg} a + e \operatorname{arctg} b = e\pi + e \operatorname{arctg}(d/c))$

Либо сумма входит в уравнение - условие задачи на описание, причем выражения a, b константные, а уровень срабатывания равен 0, либо она входит в условие задачи на преобразование, а уровень срабатывания равен 2. Если отсутствует цель "редакция", то оба выражения a, b должны быть константными, а задача долна иметь тип "преобразовать". Третий и пятый антецеденты выделены указателем "идентификатор", остальные - обрабатываются проверочными операторами.

$\forall_{ab}(0 < b \to a \operatorname{arctg} b + a \operatorname{arctg}(1/b) = a\pi/2)$

Уровень срабатывания равен 1.

$\forall_{abc}(bc - 1 = 0 \ \& \ 0 < b \to a \operatorname{arctg} b + a \operatorname{arctg} c = a\pi/2)$

Выражения b, c должны содержать квадратные радикалы (единица может возникнуть при перемножении суммы и разности двух радикалов). Первый антецедент выделен указателем "идентификатор", его левая часть обрабатывается нормализатором раскрывания скобок. Второй антецедент обрабатывается проверочным оператором. Уровень срабатывания равен 2.

Разность арктангенсов

$\forall_{abcde}(0 \le a \ \& \ 0 \le b \ \& \ c = 1 + ab \ \& \ d = a - b \to e \operatorname{arctg} a - e \operatorname{arctg} b = e \operatorname{arctg}(d/c))$

$\forall_{abcde}(a \le 0 \ \& \ b \le 0 \ \& \ c = 1 + ab \ \& \ d = a - b \to e \operatorname{arctg} a - e \operatorname{arctg} b = e \operatorname{arctg}(d/c))$

Аналогично случаю суммы.

Переход к арксинусу

$\forall_a(0 < a \to \operatorname{arctg} a = \arcsin(a/\sqrt{1+a^2}))$

Арктангенс входит в условие задачи на преобразование, имеющей цель "известны" (редактирование ответа задачи на вычисление). Выражение a константное. Условие задачи должно содержать арксинус либо арккосинус константного выражения. Уровень срабатывания равен 3.

Простейшее уравнение с арктангенсом

$\forall_{ab}(\operatorname{arctg} a = b \leftrightarrow a = \operatorname{tg} b \ \& \ \neg(\cos b = 0) \ \& \ 0 < \pi + 2b \ \& \ 0 < \pi - 2b)$

Уравнение является условием задачи на описание. Выражение a содержит неизвестные, b - не содержит. Уровень срабатывания равен 1.

Вывод следствия из уравнения с арктангенсом

$\forall_{ab}(\neg(\cos b = 0) \ \& \ \operatorname{arctg} a + b = 0 \to a + \operatorname{tg} b = 0)$

Прием имеет заголовок "выводусловия". Второй антецедент идентифицируется с условием задачи на описание. Выражения a, b содержат неизвестные. В первое из них входит тангенс либо котангенс. Уровень срабатывания равен 8.

Нормализатор общей стандартизации "нормарктангенс"

1. Арктангенс константы: $0, 1, 1/\sqrt{3}, \sqrt{3}, 2 \pm \sqrt{3}, \sqrt{2} \pm 1, 1 - \sqrt{2}, \sqrt{3} - 2, \pm\infty$
2. Минус под арктангенсом.
3. Арктангенс тангенса.
4. Арктангенс котангенса.

11.10 Прием символа "арккотангенс"

Выражение "арккотангенс(a)", прорисовываемое формульным редактором в виде $\operatorname{arcctg} a$, обозначает арккотангенс вещественного числа a. Решатель преобразует арккотангенсы в арктангенсы с помощью следующего приема:

$\forall_a(\operatorname{arcctg} a = \pi/2 - \operatorname{arctg} a)$.

Прием применяется без ограничений. Уровень срабатывания равен 0.

11.11 Приемы, используемые для редактирования параметрических описаний, возникающих в ответах тригонометрических уравнений и неравенств

Параметрические описания, возникающие в тригонометрических уравнениях

1. Сложение дробных выражений.

   $\forall_{abcdepx}(p = b/c{+}d \rightarrow \exists_n(x = a(b/c{+}d)/e \,\&\, n-$целое$) \leftrightarrow \exists_n(x = ap/e \,\&\, n-$целое$))$

   Утверждение существования является условием задачи на описание. Выражение x содержит неизвестные, правая часть равенства для x неизвестных не содержит. Допускаются вырожденные единичные значения a, e. Указатель "содержится(x14 x2 x4)" разрешает вхождения переменной n в b, d. Антецедент выделен указателем "идентификатор". Его правая часть обрабатывается нормализатором "видумножение". Уровень срабатывания равен 0.

2. Нормировка начального значения серии корней.

   $\forall_{abcdefgh}(g = cf + d \,\&\, b = f - d \rightarrow \exists_n(h = (fn - g)a\pi/e \,\&\, n -$ целое$) \leftrightarrow \exists_n(h = (fn + b)a\pi/e \,\&\, n -$ целое$))$

   Квантор существования входит в условие задачи на описание. Переменные a, e, f, g идентифицируются с натуральными константами. Антецеденты выделены указателем "программа". Прием позволяет перейти от отрицательного начального значения к положительному. Уровень срабатывания равен 0.

   $\forall_{abcdef}(a \leq b \,\&\, b = ad + e \rightarrow \exists_n(f = (an + b)\pi/c \,\&\, n -$ целое$) \leftrightarrow \exists_n(f = (an + e)\pi/c \,\&\, n -$ целое$))$

   Аналогично предыдущему, но положительное начальное значение серии делается меньшим периода a.

3. Переход к положительному периоду серии.

   $\forall_{abc}(\exists_n(c = -an\pi/b \,\&\, n -$ целое$) \leftrightarrow \exists_n(c = an\pi/b \,\&\, n -$ целое$))$

   Квантор существования входит в условие задачи на описание. Переменные a, b идентифицируются с натуральными константами. Уровень срабатывания равен 0.

   $\forall_{abcd}(\exists_n(c = -(an + b)\pi/d \,\&\, n -$ целое$) \leftrightarrow \exists_n(c = (an - b)\pi/d \,\&\, n -$ целое$))$

   $\forall_{abcd}(\exists_n(c = (-an + b)\pi/d \,\&\, n -$ целое$) \leftrightarrow \exists_n(c = (an + b)\pi/d \,\&\, n -$ целое$))$

   Переменные a, d идентифицируются с натуральными константами, переменная b - с целочисленной константой. В остальном аналогично предыдущему.

4. Переход от серии исключаемых точек к серии промежутков.

   $\forall_{abcdex}(0 < acd \rightarrow \forall_n(n-$целое $\rightarrow \neg(x = (an{+}b)c/d{+}e)) \leftrightarrow \exists_n(n-$целое $\& \ (an + b - a)c/d + e < x \,\&\, x < (an + b)c/d + e))$

   $\forall_{abcdex}(acd < 0 \rightarrow \forall_n(n-$целое $\rightarrow \neg(x = (an{+}b)c/d{+}e)) \leftrightarrow \exists_n(n-$целое $\& \ (-an + b + a)c/d + e < x \,\&\, x < (-an + b)c/d + e))$

Кванторная импликация является условием задачи на описание, причем не имеет места этап редактирования ответа. Переменная x идентифицируется с неизвестной. Условия задачи не содержат тригонометрических операций с неизвестными. Допускаются вырожденные единичные значения a, c, d и нулевые значения b, e. Антецедент обрабатывается проверочным оператором. Уровень срабатывания равен 1.

5. Пересечение серии с конечным промежутком.

   $\forall_{abcdefgh}(a \leq b \ \& \ b \leq c \ \& \ d = [(\pi e - af)/\pi g] \ \& \ h = [(cf - \pi e)/\pi g] + d + 1 \rightarrow \exists_n(b = \pi(gn + e)/f \ \& \ n - \text{целое}) \leftrightarrow \exists_i(i \in \{0, \ldots, h-1\} \ \& \ b = \pi(g(i-d)+e)/f))$

   Квантор существования входит в условие задачи на описание. Переменная b идентифицируется с неизвестной, переменные d, e, f, g, h - с десятичными константами, переменные a, c - с константными выражениями. Первые два антецедента идентифицируются с утверждениями текущего контекста, они определяют конечный промежуток изменения неизвестной b. Указатели "альтернатива" разрешают рассмотрение как строгих, так и нестрогих неравенств. Третий и четвертый антецеденты выделены указателями "идентификатор". Они вычисляют с помощью нормализаторов общей стандартизации целочисленные константы d, h. Указатель "или(...)" определяет развертку заменяющего квантора существования в дизъюнкцию. Уровень срабатывания равен 1.

   $\forall_{abcdefgh}(a \leq b \ \& \ b \leq c \ \& \ d = [(\pi e - af)/\pi g] \ \& \ h = [(cf - \pi e)/\pi g] + d + 1 \rightarrow \exists_n(b = \pi(gn + e)/f \ \& \ n - \text{целое} \ \& \ p(n)) \leftrightarrow \exists_i(i \in \{0, \ldots, h-1\} \ \& \ b = \pi(g(i-d)+e)/f \ \& \ p(i)))$

   Аналогично предыдущему, но в описании серии участвует дополнительное условие $p(n)$ на варьируемый параметр. Здесь p - функциональная переменная. Уровень срабатывания равен 2.

6. Усмотрение избыточного целочисленного параметра.

   $\forall_{abcdenkx}(n - \text{целое} \ \& \ b = ak \rightarrow \exists_m(x = e(am + bn + c)/d \ \& \ m - \text{целое}) \leftrightarrow \exists_m(x = e(am + c)/d \ \& \ m - \text{целое}))$

   Квантор существования входит в условие задачи на описание. Переменная x идентифицируется с неизвестной, переменные a, b - с целочисленными константами. Если после идентификации b остается невырожденный остаток сомножителей n, то первый антецедент, обрабатываемый проверочным оператором, убеждается, что этот остаток целочисленный. Второй антецедент, выделенный указателем "программа", проверяет, что b делится на a. В этой ситуации член bn оказывается избыточным и отбрасывается. Уровень срабатывания равен 1.

7. Усмотрение серии ограничений на значение неизвестной. Обычно кванторные условия, определяющие серию допустимых промежутков значений неизвестной либо серию отбрасываемых точек, вводятся специальными приемами и сразу помечаются комментарием "серия". Этот комментарий необходим для обращения к вспомогательным задачам, обеспечивающим доведение параметрических описаний до окончательного вида. Если аналогичные кванторные условия возникли из каких-либо иных источников, то для них комментарий "серия" создается следующими приемами:

   $\forall_{fx}(\forall_n(n - \text{целое} \rightarrow \neg(x = f(n))) \rightarrow \emptyset)$

$\forall_{fgx}(\exists_n(n - \text{целое}\ \&\ f(n) < x\ \&\ x < g(n)) \to \emptyset)$

Приемы имеют заголовок "замечание". Кванторное утверждение является условием задачи на описание. Переменная x идентифицируется с неизвестной, $f(n)$ и $g(n)$ неизвестных не содержат. Переменные f, g - функциональные. Не имеет места этап редактирования ответа. Уровень срабатывания равен 1.

Параметрические описания, возникающие в тригонометрических неравенствах

1. Изменение знака целочисленного параметра.

   $\forall_{abcdefpqrs}(ac < 0\ \&\ pr < 0 \to \exists_n(n - \text{целое}\ \&\ a(b + cn)/d \leq e\ \&\ e \leq p(q + rn)/s\ \&\ f(n)) \leftrightarrow \exists_n(n - \text{целое}\ \&\ a(b - cn)/d \leq e\ \&\ e \leq p(q - rn)/s\ \&\ f(-n)))$

   Если периоды для числителей нижней и верхней границ промежутка отрицательны, что усматривается из различия знаков коэффициентов a, c и p, r, то выполняется переход к положительным значениям этих периодов путем изменения знака параметра n. Утверждение существования, задающее серию промежутков, входит в условие задачи на описание, помеченное комментарием "серия". Переменная e идентифицируется с неизвестной. Первые два антецедента обрабатываются проверочными операторами. Указатели "альтернатива(...)" разрешают рассмотрение для границ промежутков как строгих, так и нестрогих неравенств. Переменная f - функциональная. Допускаются вырожденные единичные значения a, c, d, p, r, s и нулевые значения b, q. Уровень срабатывания равен 0.

2. Выделение слагаемого, определяющего период серии промежутков. Выражение для границы промежутка преобразуется к виду суммы константной и варьируемой частей:

   $\forall_{abcdef}(\exists_n(n - \text{целое}\ \&\ a(b+cn)/d \leq e\ \&\ f(n)) \leftrightarrow \exists_n(n - \text{целое}\ \&\ ab/d + acn/d \leq e\ \&\ f(n)))$

   $\forall_{abcdef}(\exists_n(n - \text{целое}\ \&\ e \leq a(b + cn)/d\ \&\ f(n)) \leftrightarrow \exists_n(n - \text{целое}\ \&\ e \leq ab/d + acn/d\ \&\ f(n)))$

   Утверждение существования входит в условие задачи на описание, выделенное комментарием "серия". Переменная e идентифицируется с неизвестной, не входящей в противоположную часть неравенства. Разрешается рассмотрение как нестрогого, так и строгого неравенств. Допускаются вырожденные единичные значения a, c, d, но запрещается одновременное обращение в единицу a и d. Уровень срабатывания равен 1.

3. Обращение к пакетному оператору "пересечениесерий" для пересечения двух серий промежутков. По мере того, как в условии задачи на описание появляются различные серии допустимых промежутков для одной и той же неизвестной, происходит пересечение этих серий - замена на одну новую серию либо на дизъюнкцию нескольких новых серий. Фактически вычисления производятся описываемым ниже нормализатором "пересечениесерий". Обращение к нему выполняется следующим приемом:

   $\forall_{afgpq}(a = (\exists_n(n - \text{целое}\ \&\ f(n) \leq x\ \&\ x \leq g(n))\ \&\ \exists_m(m - \text{целое}\ \&\ p(m) \leq x\ \&\ x \leq q(m))) \to \exists_n(n - \text{целое}\ \&\ f(n) \leq x\ \&\ x \leq g(n))\ \&\ \exists_m(m - \text{целое}\ \&\ p(m) \leq x\ \&\ x \leq q(m)) \leftrightarrow a)$

Заголовок приема - "заменаусловия(второйтерм)". Оба утверждения существования являются условиями задачи на описание, причем первое из них помечено комментарием "серия". Переменная x идентифицируется с неизвестной, выражения $f(n), g(n), p(m), q(m)$ неизвестных не содержат. Антецедент выделен указателем "идентификатор". Его правая часть обрабатывается нормализаторами "пересечениесерий" и "нормлог". Проверяется, что в результате исключаются конъюнкции, имеющие своим операндом квантор существования. В промежутках для x разрешаются как нестрогие, так и строгие неравенства. Заменяющее условие сопровождается комментарием "серия". Уровень срабатывания приема равен 1.

4. Нормализатор "пересечениесерий". Конъюнкция двух параметрических описаний серий промежутков преобразуется в одно параметрическое описание либо в дизъюнкцию нескольких.

    (a) Пересечение двух серий, имеющих общий период.

    $\forall_{abcdefghik}(a = [(b-c)d/ke]\ \&\ f = [(g-h)d/ke] \rightarrow \exists_m(m-\text{целое}\ \&\ emk/d+h < i\ \&\ i < emk/d + b)\ \&\ \exists_n(n - \text{целое}\ \&\ enk/d + c < i\ \&\ i < enk/d + g) \leftrightarrow \exists_j(j \in \{0, \dots, \max(a + f, 0)\}\ \&\ \exists_m(m - \text{целое}\ \&\ emk/d + h < i\ \&\ emk/d + (j - f)ek/d + c < i\ \&\ i < emk/d + b\ \&\ i < emk/d + (j - f)ek/d + g)))$

    Переменная d идентифицируется с натуральной константой, переменная e - с десятичной константой. Общий период серий промежутков равен ek/d, где k - некоторое выражение (обычно - число π). Антецеденты выделены указателем "идентификатор". Их правые части обрабатываются с помощью нормализаторов общей стандартизации и нормализатора "видумножение", причем после упрощения должны возникать десятичные константы a, f. Первая из них указывает наибольшее число периодов, на которые может понадобиться сдвинуть вправо второй промежуток, чтобы его левый конец не оказался правее правого конца первого промежутка. Вторая - наибольшее число периодов, на которые может понадобиться сдвинуть второй промежуток влево, чтобы его правый конец не оказался левее левого конца первого промежутка. Очевидно, в этих пределах и следует рассматривать те кратные периоду величины сдвига второго промежутка, при которых он может пересечься с первым. В заменяющем терме величина сдвига определяется параметром j, изменяющимся от 0 до $\max(a+f), 0$. Сама эта величина равна $(j-f)ek/d$. Указатель "или(...)" определяет развертку квантора существования по j в дизъюнкцию. Членами дизъюнкции служат параметрические описания для серий пересечений первого промежутка со сдвинутым вторым промежутком. Под квантором существования здесь собраны четыре неравенства, преобразуемых нормализатором "нормпромежутка" (см. ниже) в два неравенства, равенство либо в константу "ложь". Указатели "альтернатива" разрешают обработку промежутков с произвольными комбинациями строгих и нестрогих неравенств. Допускаются вырожденные единичные значения переменных d, e, k и нулевые значения b, c, g, h. Уровень срабатывания равен 1.

    (b) Преобразование двух серий к общему периоду.

    $\forall_{abcdefijklmr}(m = \text{нод}(kh, ij)\ \&\ me = ij\ \&\ mf = hk \rightarrow \exists_n(hrn/i + a \leq l\ \&\ l \leq hrn/i + b\ \&\ n - \text{целое})\ \&\ \exists_p(jrp/k + cl \leq l\ \&\ l \leq jrp/k + d\ \&\ p - \text{целое}) \leftrightarrow \exists_{gq}(g \in \{0, \dots, e - 1\}\ \&\ q \in \{0, \dots, f - 1\}\ \&\ (\exists_n(ehrn/i + hrg/i + a \leq$

l & $l \leq ehrn/i + hrg/i + b$ & n – целое) & $\exists_p(jfrp/k + jrq/k + c \leq l$ & $l \leq jfrp/k + jrq/k + d$ & p – целое))))

Переменные h, i, j, k идентифицируются с натуральными константами. Периоды исходных серий равны $hr/i, jr/k$. Антецеденты выделены указателями "программа". Они находят такие натуральные множители e, f, что e - кратный первый период равен f - кратному второму. Проверяется, что либо e, либо f отлично от 1. Заменяющий терм преобразует серии к общему периоду $fjr/k = ehr/i$. При этом учитываются значения $g = 0, \ldots, e-1$ возможного сдвига первого промежутка внутри нового периода и значения $q = 0, \ldots, f-1$ сдвига внутри этого периода второго промежутка. Квантор существования по g, q разворачивается в дизъюнкцию конъюнкций соответствующих серий. Каждая конъюнкция рекурсивным образом обрабатывается нормализатором "пересечениесерий". В исходных сериях промежутков разрешаются любые комбинации строгих и нестрогих неравенств. Допускаются вырожденные единичные значения h, i, j, k, r и нулевые значения a, b, c, d. Уровень срабатывания равен 2.

5. Нормализатор "нормпромежутка". Обрабатывается конъюнкция неравенств, задающих условие принадлежности точки двум пересекаемым промежуткам.

   (a) Конъюнкция двух односторонних неравенств.

   $\forall_{abc}(0 < a - b \to a \leq c \,\&\, b \leq c \leftrightarrow a \leq c)$

   $\forall_{abc}(0 < a - b \to c \leq a \,\&\, c \leq b \leftrightarrow c \leq b)$

   Переменная c идентифицируется с переменной. Указатели "альтернатива(...)" разрешают рассматривать случай строгого отбрасываемого неравенства и строгого сохраняемого неравенства. Антецедент обрабатывается проверочным оператором. Уровень срабатывания равен 1.

   $\forall_{abc}(0 \leq a - b \to a \leq c \,\&\, b \leq c \leftrightarrow a \leq c)$

   $\forall_{abc}(0 \leq a - b \to c \leq a \,\&\, c \leq b \leftrightarrow c \leq b)$

   Аналогично предыдущему. Указатель "альтернатива(...)" позволяет рассматривать как двух нестрогих, так и двух строгих неравенства. Уровень срабатывания равен 2.

   $\forall_{ab}(a \leq b \,\&\, a < b \leftrightarrow a < b)$

   Уровень срабатывания равен 1.

   (b) Усмотрение пустого пересечения.

   $\forall_{ab}(\neg(a \leq b \,\&\, b < a))$

   Первое неравенство может быть строгим. Конъюнкция заменяется приемом на константу "ложь".

   $\forall_{abc}(0 < b - c \to \neg(b \leq a \,\&\, a \leq c))$

   Антецедент обрабатывается проверочным оператором. Каждое неравенство под отрицанием может быть как нестрогим, так и строгим. Выражение a должно быть неконстантным. Уровень срабатывания приемов равен 1.

   (c) Пересечение сводится к единственной точке.

   $\forall_{ab}(a \leq b \,\&\, b \leq a \leftrightarrow a = b)$

   Уровень срабатывания равен 1.

6. Пересечение серии промежутков с промежутком.

   $\forall_{abcdefpqx}(e \leq x \ \& \ x \leq f \ \& \ p = [(bd - be)/a] \ \& \ q = [(bf - bc)/a] + p + 1 \ \& \ 0 < a \ \& \ 0 < b \rightarrow \exists_n(n - \text{целое} \ \& \ an/b + c \leq x \ \& \ x \leq an/b + d) \leftrightarrow \exists_i(i \in \{0, \ldots, q - 1\} \ \& \ a(i - p)/b + c \leq x \ \& \ x \leq a(i - p)/b + d))$

   Квантор существования входит в условие задачи на описание. Переменная x идентифицируется с неизвестной. Первые два антецедента идентифицируются с другими условиями задачи, они определяют конечный промежуток $[e, f]$ изменения неизвестной. Третий и четверый антецеденты выделены указателем "идентификатор". Они определяют наименьшее допустимое значение $-p$ параметра серии n, для которого возможно пересечение, и число q последовательных промежутков серии, пересекаемых с промежутком $[e, f]$. Целые части обрабатываются вспомогательными задачами на преобразование. Проверяется, что результаты p, q суть десятичные константы. Два последних антецедента обрабатываются проверочными операторами. Указатель "или(...)" определяет развертку заменяющего квантора существования в дизъюнкцию. Указатели "альтернатива" обеспечивают рассмотрение как нестрогих, так и строгих неравенств. Допускаются вырожденные единичные значения a, b и нулевые значения c, d. Уровень срабатывания равен 1.

7. Определение связи между двумя различными целочисленными параметрами для пересекаемых промежутков.

   $\forall_{abcdeghijkmn}(m - \text{целое} \ \& \ n - \text{целое} \ \& \ abm/c + d < i \ \& \ i \leq abm/c + e \ \& \ abn/c + d \leq i \ \& \ i \leq abn/c + h \ \& \ j = [(h - d)c/ab] \ \& \ k = [(e - g)c/ab] \rightarrow \exists_f(f \in \{0, \ldots, j + k\} \ \& \ n = m - j + f))$

   $\forall_{abcdeghijkmn}(m - \text{целое} \ \& \ n - \text{целое} \ \& \ abm/c + d \leq i \ \& \ i \leq abm/c + e \ \& \ abn/c + d \leq i \ \& \ i \leq abn/c + h \ \& \ j = [(h - d)c/ab] \ \& \ k = [(e - g)c/ab] \rightarrow \exists_f(f \in \{0, \ldots, j + k\} \ \& \ n = m - j + f))$

   Приемы имеют заголовок "выводусловия". Антецеденты с третьего по шестой идентифицируются с условиями задачи на описание, имеющей переменную i своей неизвестной. Переменные a, c идентифицируются с натуральными константами. Задача имеет цель (серия $y_1 \ldots y_k$), причем m, n - различные переменные списка $y_1, \ldots, y_k$. Это означает, что предпринимается редактирование списка утверждений некоторого параметрического описания по переменным $y_1, \ldots, y_k$. Периоды пересекаемых серий промежутков для i совпадают. Должны отсутствовать дизъюнктивные условия, а также равенства для m, n. Первые два антецедента обрабатываются проверочными операторами. Последние два антецедента выделены указателем "идентификатор". Они определяют наименьшее смещение $-j$ параметра n по отношению к m, а также наибольшее смещение k. Выводимый квантор существования разворачивается в дизъюнкцию равенств, связывающих между собой параметры n, m. Эта дизъюнкция сопровождается комментарием "разборслучаев". Два приема созданы из-за того, что в одном случае точкой привязки служит вхождение символа "меньше", а в другом - символа "меньшеилиравно". Указатели "альтернатива" обеспечивают произвольные комбинации нестрогих и строгих неравенств для остальных антецедентов. Уровень срабатывания равен 0.

8. Разбор случаев для согласования периодов двух серий промежутков.

$\forall_{abcdefghijpqrst}(\text{нод}(ah, bf) = r \;\&\; ah = rs \;\&\; bf = rt \;\&\; apj/b + c < q \;\&\; q \leq apj/b + d \;\&\; fgj/h + e \leq q \;\&\; q \leq fgj/h + i \rightarrow \exists_{kl}(k \in \{0, \ldots, s-1\} \;\&\; l \in \{0, \ldots, t-1\} \;\&\; \exists_{mn}(m - \text{целое} \;\&\; n - \text{целое} \;\&\; p = tm + l \;\&\; g = sn + k)))$

$\forall_{abcdefghijpqrst}(\text{нод}(ah, bf) = r \;\&\; ah = rs \;\&\; bf = rt \;\&\; apj/b + c \leq q \;\&\; q \leq apj/b + d \;\&\; fgj/h + e \leq q \;\&\; q \leq fgj/h + i \rightarrow \exists_{kl}(k \in \{0, \ldots, s-1\} \;\&\; l \in \{0, \ldots, t-1\} \;\&\; \exists_{mn}(m - \text{целое} \;\&\; n - \text{целое} \;\&\; p = tm + l \;\&\; g = sn + k)))$

Приемы имеют заголовок "выводусловия". Последние четыре антецедента идентифицируются с условиями задачи на описание, имеющей переменную q своей неизвестной. Переменные a, b, f, h идентифицируются с натуральными константами. Задача имеет цель (серия ...), в которой упоминаются различные переменные p, g. Должны отсутствовать дизъюнктивные условия и условия существования. Первые три антецедента выделены указателем "программа". Здесь вычисляются параметры s, t, домножение на которые исходных периодов дает их наименьшее общее кратное. Выводимый квантор существования разворачивается в дизъюнкцию параметрических описаний, выражающих исходные параметры p, g через новые параметры m, n. В каждом подслучае получаются две серии промежутков, имеющие одинаковый период. Дизъюнкция сопровождается комментариями "серия" и "разборслучаев". Уровень срабатывания равен 1.

9. Выделение в неравенстве для серии промежутков слагаемого, задающего период серии.

   $\forall_{abcdef}((ad + b)e/c + f = ade/c + be/c + f)$

   Прием стандартизирует неравенства таким образом, чтобы подготовить возможность срабатывания приемов из двух предыдущих пунктов. Сумма является одной из частей неравенства - условия задачи на описание. Она не содержит неизвестных. Другой частью неравенства должна являться неизвестная. Задача имеет цель (серия ...). d - переменная, упоминаемая в этой цели. Переменная a идентифицируется с десятичной константой, переменная b - с целочисленной, а переменная c - с натуральной. Уровень срабатывания равен 2.

Глава 12

Приемы, связанные с гиперболическими функциями

Гиперболические функции при обучении решателя рассматривались редко, и приемов для них создано немного.

12.1 Приемы символа "гипсинус"

Выражение "гипсинус(a)", прорисовываемое формульным редактором в виде $\operatorname{sh} a$, обозначает гиперболический синус вещественного числа a. Справочники "тип", "арность", "одз", "типданных" характеризуют простейшие свойства символа "гипсинус".

1. $\operatorname{sh} 0 = 0$
2. $\forall_a(\operatorname{sh} a = 0 \leftrightarrow a = 0)$
3. $\forall_a(\operatorname{sh}(-a) = -\operatorname{sh} a)$

   Уровень срабатывания приемов равен 0.

4. Выражение через экспоненту.

   $\forall_x(\operatorname{sh} x = (\exp x - \exp(-x))/2)$

   На этой теореме созданы два приема. Первый применяется в условиях задач на описание. При решении дифференциальных уравнений, варьируемые параметры которых входят в x, допускается уровень срабатывания 1. На уровне 2 прием срабатывает, если x содержит неизвестные. Таким образом, при решении уравнений обычно происходит быстрое исключение гиперболических синусов. Второй прием применяется в условиях задач на преобразование. Здесь требуется, чтобы условие уже содержало экспоненту с основанием e и показателем, имеющим общий параметр с x. Не должен иметь место этап редактирования ответа. Уровень срабатывания равен 6.

5. Переход к двойному аргументу.

   $\forall_{abc}(c(\operatorname{sh} a)^b(\operatorname{ch} a)^b = c(\operatorname{sh}(2a))^b/2^b)$

   Произведение входит в условие задачи на преобразование. Для случая упрощения подынтегрального выражения создан другой прием (см. ниже). Поэтому данный прием при преобразовании подынтегральных выражений блокируется. Уровень срабатывания равен 1.

6. Сумма квадрата гиперболического синуса и единицы.

   $\forall_{ab}(a(\operatorname{sh} b)^2 + a = a(\operatorname{ch} b)^2)$

   Уровень срабатывания равен 3.

7. Знак гиперболического синуса.

   $\forall_b(b < 0 \leftrightarrow \operatorname{sh} b < 0)$

   $\forall_b(0 < b \leftrightarrow 0 < \operatorname{sh} b)$

   Направление замены - справа налево. Уровень срабатывания равен 2.

8. Преобразование подынтегрального выражения к виду суммы.

   $\forall_{abcdep}(p = a(\operatorname{sh} b)^c/d + e \rightarrow a(\operatorname{sh} b)^c/d + e = p)$

   $\forall_{abcdep}(p = a \operatorname{sh} b \operatorname{sh} c/d + e \rightarrow a \operatorname{sh} b \operatorname{sh} c/d + e = p)$

   Выражение входит в условие задачи на преобразование, решаемой для упрощения подынтегрального выражения. Переменная интегрирования входит в аргументы гиперболических синусов и не встречается в знаменателе d.Она также не должна встречаться в подынтегральном выражении под логарифмом, арксинусом, арккосинусом и арктангенсом. Показатель степени идентифицируется с натуральной константой. Каждое вхождение в подынтегральное выражение гиперболической функции, аргумент которой содержит переменную интегрирования, должно быть "алгебраическим" - надоперациями могут быть только $+, -, \times, /, =, <, \leq$, модули и степени. В последнем случае вхождение должно быть расположено внутри основания степени. Уровень срабатывания равен 2.

9. Переход к двойному аргументу при интегрировании.

   $\forall_{abc}(c(\operatorname{sh} a)^b(\operatorname{ch} a)^b = c(\operatorname{sh}(2a))^b/2^b)$

   $\forall_{abc}(c|\operatorname{sh} a|^b(\operatorname{ch} a)^b = c|\operatorname{sh}(2a)|^b/2^b)$

   Выражение входит в условие задачи на преобразование, решаемой для упрощения подынтегрального выражения. Переменная интегрирования входит в a. Проверяется, что в рассматриваемом выражении все вхождения гиперболического аргумента a естественным образом преобразуются к двойному аргументу. Проверяется также отсутствие гиперболических аргументов, отличных от $a, 2a$. Уровень срабатывания равен 1.

10. Нормализатор общей стандартизации "нормгипсинус".

    В нормализаторе два приема - $\operatorname{sh} 0 = 0$ и $\forall_a(\operatorname{sh}(-a) = -\operatorname{sh} a)$.

12.2 Приемы символа "гипкосинус"

Выражение "гипкосинус(a)", прорисовываемое формульным редактором в виде $\operatorname{ch} a$, обозначает гиперболический косинус вещественного числа a. Приемы этого символа аналогичны приемам символа "гипсинус".

1. $\operatorname{ch} 0 = 1$

2. $\operatorname{ch}(-a) = \operatorname{ch} a$

   Уровень срабатывания равен 0.

3. Выражение через экспоненту.

   $\forall_x(\operatorname{ch} x = (\exp x + \exp(-x))/2)$

   Два приема, аналогичных случаю гиперболического синуса.

4. Разность квадратов гиперболических синуса и косинуса.

   $\forall_{ab}(a(\operatorname{ch} b)^2 - a(\operatorname{sh} b)^2 = a)$

   Уровень срабатывания равен 1.

5. Сумма квадратов гиперболических косинуса и синуса.

   $\forall_{ab}(a(\operatorname{sh} b)^2 + a(\operatorname{ch} b)^2 = a \operatorname{ch}(2b))$

   Сумма входит в условие задачи на преобразование. Уровень срабатывания равен 3.

6. Раскрывание скобок для умножения гиперболического тангенса на косинус.

   $\forall_{abmnx}((a(\operatorname{th} x)^m + b)(\operatorname{ch} x)^n = a(\operatorname{th} x)^m(\operatorname{ch} x)^n + b(\operatorname{ch} x)^n)$

   Переменные m, n идентифицируются с натуральными константами. Уровень срабатывания равен 3.

7. Разность гиперболических синуса и косинуса.

   $\forall_{ab}(b \operatorname{ch} a - b \operatorname{sh} a = b \exp(-a))$

   Уровень срабатывания равен 1.

8. Сумма гиперболических синуса и косинуса.

   $\forall_{ab}(b \operatorname{ch} a + b \operatorname{sh} a = b \exp a)$

   Уровень срабатывания равен 1.

9. Преобразование подынтегрального выражения к виду суммы.

   $\forall_{abcdep}(p = a(\operatorname{ch} b)^c/d + e \rightarrow a(\operatorname{ch} b)^c/d + e = p)$

   $\forall_{abcdep}(p = a \operatorname{ch} b \operatorname{ch} c/d + e \rightarrow a \operatorname{ch} b \operatorname{ch} c/d + e = p)$

   Аналогично случаю гиперболического синуса.

10. Нормализатор общей стандартизации "нормгипкосинус". В нормализаторе два приема - $\operatorname{ch} 0 = 1$ и $\forall_a(\operatorname{ch}(-a) = \operatorname{ch} a)$.

12.3 Приемы символа "гиптангенс"

Выражение "гиптангенс(a)", прорисовываемое формульным редактором в виде $\operatorname{th} a$, обозначает гиперболический тангенс вещественного числа a.

1. Выражение через экспоненту.

   $\forall_a(\operatorname{th} a = (\exp(2a) - 1)/(exp(2a) + 1))$

   По теореме созданы три приема. Первый - для условий задач на преобразование - аналогичен случаям гиперболических синуса и косинуса. Второй - для условий задач на описание - срабатывает на уровне 2 при неизвестном a. Наконец, третий прием срабатывает на уровне 0 внутри описателей "класс", расположенных в посылках.

2. Произведение гиперболических тангенса и косинуса.

   $\forall_{abcdef}(a = \min(b, c)\ \&\ d = b{-}a\ \&\ e = c{-}a \rightarrow (\operatorname{th} f)^b(\operatorname{ch} f)^c = (\operatorname{th} f)^d(\operatorname{sh} f)^a(\operatorname{ch} f)^e)$

   Переменные b, c идентифицируются с натуральными константами. Антецеденты выделены указателем "программа". Уровень срабатывания равен 3.

3. Произведение гиперболического косинуса на сумму с гиперболическим тангенсом.

   $\forall_{abcn}((a(\operatorname{th} b)^n + c)(\operatorname{ch} b)^n = a(\operatorname{sh} b)^n + c(\operatorname{ch} b)^n)$

   Замена выполняется в условии задачи на преобразование, имеющей цель "упростить". Уровень срабатывания равен 3.

4. Усмотрение экспоненты.

   $\forall_{ab}((\operatorname{th} b + 1)/(a - a \operatorname{th} b) = \exp(2b)/a)$

   Уровень срабатывания равен 1.

5. Усмотрение квадрата гиперболического тангенса половинного аргумента.

   $\forall_{abcn}(a(\operatorname{ch} b - 1)^n/c(\operatorname{ch} b + 1)^n = a(\operatorname{th}(b/2))^2/c)$

   Замена выполняется в условии задачи на преобразование, имеющей цель "упростить". Уровень срабатывания равен 4.

6. Усмотрение гиперболического синуса двойного аргумента.

   $\forall_{abnx}(a(\operatorname{th} x)^n/b(1 - (\operatorname{th} x)^2)^n = a(\operatorname{sh}(2x))^n/b2^n)$

   Уровень срабатывания равен 2.

7. Разность квадрата гиперболического тангенса и единицы.

   $\forall_{ab}(a - a(\operatorname{th} b)^2 = a/(\operatorname{ch} b)^2)$

   Уровень срабатывания равен 2.

8. Логарифм суммы с гиперболическим тангенсом.

   $\forall_a(\ln(\operatorname{th} a + 1) = a - \ln \operatorname{ch} a)$

   $\forall_a(\ln(1 - \operatorname{th} a) = -a - \ln \operatorname{ch} a)$

   Уровень срабатывания равен 1.

   $\forall_{ab}(\ln(a(\operatorname{th} x)^2 + b \operatorname{th} x + a) = \ln(a \operatorname{ch}(2x) + b \operatorname{sh}(2x)/2) - 2 \ln \operatorname{ch} x)$

   Уровень срабатывания равен 3.

9. Равенство разности гиперболических тангенсов нулю.

   $\forall_{abc}(\neg(a = 0) \to a \operatorname{th} b - a \operatorname{th} c = 0 \leftrightarrow b - c = 0)$

   Уровень срабатывания равен 1.

10. Нормализатор общей стандартизации "нормгиптангенс". В нормализаторе два приема - $\operatorname{th} 0 = 0$ и $\forall_a(\operatorname{th}(-a) = -\operatorname{th} a)$.

12.4 Приемы символа "гипкотангенс"

Выражение "гипкотангенс(a)", прорисовываемое формульным редактором в виде $\operatorname{cth} a$, обозначает гиперболический котангенс вещественного числа a.

1. Выражение через экспоненту.

   $\forall_a(\neg(a = 0) \to \operatorname{cth} a = (\exp(2a) + 1)/(\exp(2a) - 1))$

   Аналогично случаю гиперболического тангенса, но созданы лишь 2 приема - для условий задач на преобразование и описание.

2. Произведение гиперболических котангенса и синуса.

   $\forall_{abcdef}(d = \min(b, c) \mathbin{\&} e = b{-}d \mathbin{\&} f = c{-}d \to (\operatorname{cth} a)^b(\operatorname{sh} a)^c = (\operatorname{cth} a)^e(\operatorname{ch} a)^d(\operatorname{sh} a)^f)$

   Переменные b, c идентифицируются с натуральными константами. Уровень срабатывания равен 3.

3. Переход от гиперболического тангенса к гиперболическому котангенсу.

   $\forall_{abc}(a/b \operatorname{th} c = a \operatorname{cth} c/b)$

   Уровень срабатывания равен 1.

4. Вынесение минуса.

   $\forall_a(\operatorname{cth}(-a) = -\operatorname{cth} a)$

   Уровень срабатывания равен 0.

5. Нормализатор общей стандартизации "нормгипкотангенс". В нормализаторе всего один прием - $\forall_a(\operatorname{cth}(-a) = -\operatorname{cth} a)$.

Глава 13

Приемы, связанные с целыми и рациональными числами

13.1 Приемы символа "целое"

Утверждение "целое(a)", прорисовываемое формульным редактором в виде "a – целое", означает, что a есть целое вещественное число. Символ характеризуется справочниками "арность", "предикатныйсимвол", "род", "родобъекта". Он задает один из основных типов объектов, причем надтипами являются типы "рациональное", "число", "комплексное".

Общая стандартизация условия целочисленности

1. Усмотрение целого числа. Прием с псевдотеоремой "родобъекта(целое)" и заголовком "родобъекта" обеспечивает замену на логическую константу "истина" тех утверждений "целое(t)", у которых заголовком выражения t служит символ операции либо константа, имеющие, согласно справочнику "тип", заведомо целочисленные значения. Уровень срабатывания равен 0. Прием с теоремой $\forall_a(a - \text{целое} \to a - \text{целое})$ и заголовком "второйтерм" обеспечивает замену на логическую константу "истина" тех утверждений "целое(a), у которых целочисленность выражения a распознается проверочным оператором "усмцелое". Если утверждение является посылкой задачи на исследование и a - неизвестная, то преобразование блокируется. Уровень срабатывания равен 1. Наконец, прием с теоремой $\forall_a(a - \text{натуральное} \to a - \text{целое})$ заменяет на константу "истина" утверждение "целое(a)", если в контексте содержится утверждение "натуральное(a)". Уровни срабатывания равны 1 и 3.

2. Усмотрение нецелого числа.

   $\forall_a(\neg(a - \text{целое}) \to \neg(a - \text{целое}))$

   Прием имеет заголовок "второйтерм". Антецедент обрабатывается проверочным оператором "усмнецелое". Если утверждение входит в условие задачи на описание, содержащее неизвестные, то уровень срабатывания равен 4, иначе - равен 2.

   $\forall_a(a - \text{число}\ \&\ \neg(a = 0)\ \&\ b - \text{число} \to \neg(ai + b - \text{целое}))$

   Здесь операции сложения и умножения - комплекснозначные. Антецеденты обрабатываются проверочными операторами. Допускаются вырожденные единичное значение a и нулевое значение b. Уровень срабатывания равен 1.

$\forall_{ab}(\neg(b - \text{целое})\ \&\ a - \text{целое} \to \neg(a = b))$

Прием имеет заголовок "второйтерм". Выражение b константное. Первый антецедент обрабатывается проверочным оператором, второй - берется в контексте. Уровень срабатывания равен 3.

3. Минус.

   $\forall_n((-n) - \text{целое} \leftrightarrow n - \text{целое})$

   Уровень срабатывания равен 2.

4. Модуль.

   $\forall_a(a - \text{целое} \leftrightarrow |a| - \text{целое})$

   Замена выполняется справа налево. Уровень срабатывания равен 2.

5. Сумма.

   $\forall_{mn}(m - \text{целое} \to (m + n) - \text{целое} \leftrightarrow n - \text{целое})$

   Антецедент обрабатывается проверочным оператором. Уровень срабатывания равен 1.

6. Произведение.

   $\forall_{mn}(m - \text{целое}\ \&\ n - \text{целое} \to mn - \text{целое})$

   Прием имеет заголовок "второйтерм". Антецеденты обрабатываются проверочными операторами. Уровень срабатывания равен 2.

7. Дробь.

   $\forall_{mn}(m - \text{целое}\ \&\ n - \text{целое} \to (m/n) - \text{целое} \leftrightarrow n|m)$

   Антецеденты обрабатываются проверочными операторами. Уровень срабатывания равен 2.

   $\forall_{mn}(n - \text{целое} \to (m/n) - \text{целое} \leftrightarrow n|m\ \&\ m - \text{целое})$

   Антецедент обрабатывается проверочным оператором. Текущая задача имеет тип "описать". Уровень срабатывания равен 2.

8. Кратное пи.

   $\forall_m(m - \text{целое} \to \pi m - \text{целое} \leftrightarrow m = 0)$

   Антецедент обрабатывается проверочным оператором. Уровень срабатывания равен 0.

9. Выделение слагаемого числителя, кратного знаменателю.

   $\forall_{abc}(b - \text{целое} \to (ab + c)/a - \text{целое} \leftrightarrow (c/a) - \text{целое})$

   Антецедент обрабатывается проверочным оператором. Уровень срабатывания равен 1.

10. Переход к параметрическому описанию.

    $\forall_{abn}(n - \text{целое} \to an/b - \text{целое} \leftrightarrow n = 0 \ \vee \ a/b - \text{rational} \ \& \ \exists_k(k - \text{целое} \ \& \ n = k \cdot \text{знаменатель}(a/b)))$

    Преобразуемое выражение входит в условие задачи. Антецедент берется в контексте. Допускаются вырожденные единичные значения a, b, однако хотя бы одно из них должно быть отлично от 1. Если b есть единица, то не должна усматриваться целочисленность выражения a. Уровень срабатывания равен 3.

11. Дробь с константным числителем, знаменатель которой стремится к бесконечности.

    $\forall_{abcdfx}(x \to a\backslash b \ \& \ c = \lim_{x \to a\backslash b} f(x) \ \& \ (c = \infty \ \vee \ c = -\infty) \to d/f(x) - \text{целое} \leftrightarrow d = 0)$

    Первый антецедент берется в контексте, остальные выделены указателем "идентификатор": сначала с помощью нормализатора "нормпредел"вычисляется предел c, затем он сравнивается с символами бесконечности. Переменная f - функциональная. Уровень срабатывания равен 2.

12. Замена условия целочисленности на условие натуральности.

    $\forall_n(n - \text{целое} \ \& \ 1 \leq n \leftrightarrow n - \text{натуральное})$

    Заголовок приема - "второйтерм". Преобразуемая конъюнкция не расположена под описателем "отображение". Уровень срабатывания равен 1. На этой же теореме создан еще один прием, имеющий заголовок "заменаусловия(второйтерм)". Он преобразует пару условий задачи на описание при редактировании ответа. Уровень срабатывания равен 2.

    $\forall_m(0 < m \to m - \text{целое} \leftrightarrow m - \text{натуральное})$

    Утверждение входит в условие задачи на описание, находящейся на этапе редактирования ответа. Оно не расположено под кванторами и описателями. Антецедент обрабатывается проверочным оператором. Уровень срабатывания приема равен 2.

    $\forall_P(0 \leq x - 1 \to \text{set}_{xy}(x - \text{целое} \ \& \ P(x, y)) \leftrightarrow \text{set}_{xy}(x - \text{натуральное} \ \& \ P(x, y)))$

    Переменная P - функциональная. Антецедент обрабатывается проверочным оператором, причем вводится дополнительная посылка $P(x, y)$. Переменная y выделена указателем "кортежпеременных", т.е. обозначает произвольную (возможно, пустую) остаточную часть кванторной приставки. Уровень срабатывания равен 1.

Усмотрение ложности равенства целого и нецелого

$\forall_{ab}(a - \text{целое} \ \& \ \neg(b - \text{целое}) \to \neg(a = b))$

Прием имеет заголовок "второйтерм". Равенство входит в условие задачи на описание, имеющей посылку либо условие вида "целое(x)" или "натуральное(x)", где x - переменная, встречающаяся в a. Антецеденты обрабатываются проверочными операторами. Уровень срабатывания равен 2.

Исключение квантора

$\exists_n(n - \text{целое})$

$\neg(\forall_n(\neg(n - \text{целое}))$

Приемы имеют заголовок "второйтерм"и срабатывают на уровне 0.

$\forall_{an}(\exists_x(x - \text{целое}\ \&\ \forall_i(i \in \{1, \ldots, n\} \to \neg(x = a(i)))))$

Прием имеет заголовок "второйтерм". Переменная a функциональная. Указатель "развертка" определяет идентификацию кванторной импликации как конъюнкции утверждений $\neg(x = a(1)), \ldots, \neg(x = a(n))$. Указатель "внешнийквантор" блокирует попытки идентификации внешнего квантора существования через отрицание квантора общности. Уровень срабатывания равен 1.

$\forall_{abn}(\exists_c(a = b + c\ \&\ c - \text{целое}) \leftrightarrow \text{дробнаячасть}(a) = \text{дробнаячасть}(b))$

Должен иметься внешний квантор либо описатель, связывающий какую-либо переменную преобразуемого квантора существования. Как и выше, введен указатель "внешнийквантор". Уровень срабатывания равен 3.

Исключение дизъюнкции

$\forall_{mn}(m - \text{целое} \to \neg(n = m)\ \&\ n - \text{целое}\ \vee\ n = m \leftrightarrow n - \text{целое})$

Антецедент обрабатывается проверочным оператором. Уровень срабатывания приема равен 1.

Уравнения в целых числах

При решении целочисленных уравнений часто применяется разбор случаев по конечному множеству возможных значений неизвестной. Некоторые из приемов, инициирующих такой разбор случаев, уже рассматривались выше (например, прием разбора случаев по элементам целочисленного отрезка, заданного двумя нестрогими неравенствами). Кроме того, при наличии единственной неизвестной будет применяться большинство обычных "нецелочисленных" приемов решения уравнений. Ниже приводим используемые решателем специальные приемы решения целочисленных уравнений.

1. Усмотрение промежутка значений целочисленной неизвестной из неотрицательности дискриминанта квадратного уравнения.

   $\forall_{abcdex}(d = (0 \le b^2 - 4a(c - e))\ \&\ ax^2 + bx + c = e \to d)$

   Прием имеет заголовок "выводусловия". Второй антецедент идентифицируется с условием задачи на описание, имеющей не менее двух неизвестных. Коэффициенты a, b, c должны содержать единственную неизвестную y, причем задача должна иметь условие "натуральное(y)" либо "целое(y)". Выражение x должно содержать неизвестные. Первый антецедент выделен указателем "идентификатор". Входящий в него дискриминант обрабатывается нормализатором разложения на множители "видумножение", а вся правая часть - нормализатором решения нестрогих неравенств "уравнменьшеилиравно". Проверяется, что результат d имеет своими конъюнктивными членами два неравенства $p \le y, y \le q$. Уровень срабатывания равен 3.

2. Линейное целочисленное уравнение с двумя неизвестными.

   $\forall_{abcpqxy}(x -$ целое $\& \ y -$ целое $\& \ |a| \leq |b| \ \& \ b = ap + q \rightarrow ax + by = c \leftrightarrow \exists_z(x = z - py \ \& \ z -$ целое $\& \ az + qy = c))$

   Прием имеет заголовок "второйтерм". Уравнение является условием задачи на описание. Выражения x, y содержат целочисленные неизвестные, выражения a, b, c суть целочисленные константы. Запрещается наличие в x, y степенных подвыражений с неизвестными. Первые два антецедента обрабатываются проверочными операторами, два последних антецедента выделены указателем "программа". Последний из них выполняет деление с остатком b на a. В результате получается уравнение относительно y и параметра z, аналогичное исходному, но с меньшим по модулю коэффициентом при y. Преобразованное уравнение сопровождается комментарием "серия". Для дальнейшего его решения будет вводиться вспомогательная задача относительно неизвестных x, z. Это обеспечивается стандартными механизмами редактирования параметрических описаний. Уровень срабатывания приема равен 3.

3. Сокращение обеих частей целочисленного уравнения на общий множитель.

   $\forall_{abcmnk}(a + b = mn \ \& \ c = mk \ \& \ \neg(m = 0) \rightarrow a + b = c \leftrightarrow n = k)$

   Уравнение является условием задачи на описание, содержащим целочисленную неизвестную. Первые два антецеденты выделены указателем "идентификатор". Их левые части обрабатываются нормализатором ускоренного разложения на множители "факторизация". Проверяется, что найденный общий множитель отличен от единицы. Уровень срабатывания равен 2.

   $\forall_{abcdpqn}(c -$ целое $\& \ n -$ целое $\& \ 0 \leq n \ \& \ b = pq \ \& \ p|c \ \& \ d = c/p \ \&$ нод$(a, p) = 1 \rightarrow ap^n = b + c \leftrightarrow ap^{n-1} = q + d \ \& \ 0 \leq n - 1)$

   Уравнение является условием задачи на описание. Выражения a, b суть целочисленные константы, p - натуральная константа, n содержит целочисленную неизвестную. Первые три антецедента и условие делимости обрабатываются проверочными операторами. Четвертый и последний антецеденты выделены указателем "программа": находится частное q от деления b на p и проверяется взаимная простота чисел a, p. Предпоследний антецедент выделен указателем "идентификатор". Его правая часть обрабатывается вспомогательной задачей на упрощение. Уровень срабатывания равен 2.

4. Попытка разложить на множители левую часть целочисленного уравнения, если в правой части находится целочисленная константа. Теорема приема имеет вид:

   $\forall_{abc}(a = b \rightarrow b = c \leftrightarrow a = c)$

   Уравнение $b = c$ является условием задачи на описание, причем c - целочисленная константа. Выражение b содержит целочисленную неизвестную и имеет заголовок "плюс". Антецедент выделен указателем "идентификатор". Его правая часть обрабатывается нормализатором "видумножение", результат a имеет, после отбрасывания возможного внешнего минуса, заголовок "умножение" либо "степень". Уровень срабатывания равен 5. Прием подготавливает возможность разбора случаев по идентификации множителей выражения a с делителями числа c (см. следующий пункт).

5. Декомпозиция равенства произведения двух целочисленных выражений константе.

   $\forall_{nxy}(x$ – целое & y – целое $\rightarrow xy = n \leftrightarrow \exists_{pq}(n = pq$ & $x = p$ & $y = q))$

   Уравнение является условием задачи на описание и содержит целочисленную неизвестную. Выражения x, y содержат неизвестные, n есть целочисленная константа. Указатель "или(...)" определяет развертку заменяющего квантора существования в дизъюнкцию пар уравнений для x, y, соответствующих различным представлениям n в виде произведения двух целочисленных множителей. Эта дизъюнкция сопровождается комментарием "разборслучаев". Число натуральных делителей числа n не должно превосходить 15. Антецеденты обрабатываются проверочными операторами. Уровень срабатывания равен 6.

6. Преобразование уравнения к виду равенства произведения целочисленных выражений целочисленной константе.

   $\forall_{abcde}(a$ – целое & b – целое $\rightarrow ab + ac + bd = e \leftrightarrow (a + d)(b + c) = e + cd)$

   Уравнение является условием задачи на описание. Выражения a, b содержат неизвестные, выражения c, d, e суть целочисленные константы. Антецеденты обрабатываются проверочными операторами. Уровень срабатывания равен 4.

   $\forall_{abcdxy}(x$ – целое & y – целое & $\neg(a = 0) \rightarrow axy + bx + cy = d \leftrightarrow (ax + c)(ay + b) = ad + bc)$

   Аналогично предыдущему. Выражения x, y содержат неизвестные, выражения a, b, c, d суть целочисленные константы. Уровень срабатывания равен 5.

7. Усмотрение нулевого показателя степени из соображений делимости.

   $\forall_{abpqn}(a$ – целое & $\neg(p|b)$ & b – целое & n – целое & $0 \leq n$ & $ap^n = b \rightarrow n = 0)$

   Прием имеет заголовок "выводусловия". Выражение p есть натуральная константа. Последний антецедент идентифицируется с условием задачи на описание, остальные антецеденты обрабатываются проверочными операторами. Уровень срабатывания равен 1.

8. Параметрическое описание неизвестной из соображений четности-нечетности.

   $\forall_{abcxn}(a$ – целое & b – целое & c – целое & n – натуральное & $\neg(c$ – even) & b – even & $ax^n + b = c \rightarrow \exists_y(y$ – целое & $x = 2y + 1))$

   $\forall_{abcxn}(a$ – целое & b – целое & c – целое & n – натуральное & $\neg(a$ – even) & c – even & b – even & $ax^n + b = c \rightarrow \exists_y(y$ – целое & $x = 2y))$

   Приемы имеют заголовок "выводусловия". Последний антецедент идентифицируется с условием задачи на описание. Выражение x представляет собой целочисленную неизвестную, n - натуральная константа. Выводимое параметрическое описание сопровождается комментарием "серия". Уровень срабатывания равен 6.

9. Усмотрение невыполнимого равенства целого числа нецелому.

   $\forall_{ab}(a$ – целое & $\neg(b$ – целое$) \rightarrow \neg(a = b))$

   Прием имеет заголовок "второйтерм". Уравнение $a = b$ является условием задачи на описание, имеющей целочисленные неизвестные. Антецеденты обрабатываются проверочными операторами. Уровень срабатывания равен 2.

10. Усмотрение неотрицательности показателя целочисленной степени.

    $\forall_{abcpn}(\neg(p|a)\ \&\ b - \text{целое}\ \&\ c - \text{целое}\ \&\ n - \text{целое}\ \&\ ap^n + b = c \rightarrow 0 \leq n)$

    Прием имеет заголовок "выводусловия". Его последний антецедент идентифицируется с условием задачи на описание, содержащим целочисленную неизвестную. Выражения a, p суть целочисленные константы, показатель степени n неконстантный. Первые четыре антецедента обрабатываются проверочными операторами. Неотрицательность n не должна усматриваться из контекста с помощью проверочного оператора. Уровень срабатывания равен 3.

11. Раскрывание скобок с целочисленным параметром.

    $\forall_{abcdefg}(f = a(b+c)^d + e \rightarrow a(b+c)^d + e = g \leftrightarrow f = g)$

    Уравнение является условием задачи на описание, содержащим целочисленную неизвестную. Не имеет места этап редактирования ответа. Антецедент выделен указателем "идентификатор". Его правая часть обрабатывается нормализатором раскрывания скобок "стандплюс", а затем - нормализатором сумм с неизвестными "уравнплюс". Показатель степени d - натуральная константа, меньшая 6. Допускаются вырожденные единичные значения a, d, однако хотя бы одно из них должно отличаться от 1. Уровень срабатывания равен 4.

12. Разбор случаев для условия делимости.

    $\forall_{xn}(x - \text{целое} \rightarrow x|n \leftrightarrow \exists_{mp}(n = mp\ \&\ x = m))$

    Прием имеет заголовок "второйтерм". Утверждение $x|n$ идентифицируется с условием задачи на описание. Выражение x содержит неизвестные, выражение n есть целочисленная константа. Заменяющий квантор существования разворачивается в дизъюнкцию равенств $x = m$ по всем целочисленным делителям m числа n. Количество натуральных делителей числа n не должно превосходить 29. Антецедент обрабатывается проверочным оператором. Уровень срабатывания равен 5.

13. Разбор случаев по четности либо нечетности неизвестной.

    $\forall_{abcn}(a - \text{целое}\ \&\ b - \text{целое}\ \&\ c - \text{целое}\ \&\ x - \text{целое}\ \&\ \neg(a - \text{even})\ \&\ ax^n + b = c \rightarrow \exists_k(k - \text{целое}\ \&\ x = 2k)\ \vee\ \exists_k(k - \text{целое}\ \&\ x = 2k+1))$

    Прием имеет заголовок "выводусловия". Его последний антецедент идентифицируется с условием задачи на описание. Остальные антецеденты обрабатываются проверочными операторами. Переменная x идентифицируется с неизвестной, n - с натуральной константой. Из контекста либо усматривается, что c четное, либо усматривается, что оно нечетное. Однако, не усматривается ни четность, ни нечетность x. Выводимая дизъюнкция сопровождается комментариями "серия" и "разборслучаев". Уровень срабатывания равен 7.

14. Разбор случаев по ненулевому показателю степени.

    $\forall_{abpmnq}(a - \text{целое}\ \&\ b - \text{целое}\ \&\ q - \text{целое}\ \&\ m - \text{целое}\ \&\ n - \text{целое}\ \&\ 0 \leq m\ \&\ 0 \leq n\ \&\ ap^m = bp^n + q \rightarrow n = 0\ \vee\ 0 \leq n - 1)$

    Прием имеет заголовок "выводусловия". Последний антецедент идентифицируется с условием задачи на описание, содержащим целочисленную неизвестную.

Остальные антецеденты обрабатываются проверочными операторами. Показатель степени n неконстантный, причем из контекста не очевидна его неотрицательность. Выводимая дизъюнкция сопровождается комментарием "разборслучаев". Уровень срабатывания равен 3.

15. Условие целочисленности выражения с неизвестными.

    (a) Условие целочисленности радикала.

    $\forall_{abcp}(a - \text{целое}\ \&\ b - \text{целое}\ \&\ c - \text{целое}\ \&\ p - \text{целое}\ \&\ \neg(p = 0)\ \&\ (a + p\sqrt{b})/c - \text{целое} \rightarrow \sqrt{b} - \text{целое})$

    Прием имеет заголовок "выводусловия". Последний антецедент идентифицируется с условием задачи на описание, остальные - обрабатываются проверочными операторами. Выражение b содержит неизвестные. Задача имеет целочисленные неизвестные. Целочисленность радикала $\sqrt{b}$ не должна усматриваться из контекста с помощью проверочного оператора. Уровень срабатывания равен 3.

    $\forall_{ab}(a - \text{целое}\ \&\ b - \text{целое}\ \&\ 0 < b\ \&\ \sqrt{a^2 - b} - \text{целое} \rightarrow 2|a| - 1 \leq b)$

    Прием имеет заголовок "выводусловия". Последний антецедент идентифицируется с условием задачи на описание, содержащим неизвестные. Остальные антецеденты обрабатываются проверочными операторами. Уровень срабатывания равен 4.

    $\forall_{ab}(\sqrt{a}\sqrt{b} - \text{целое} \leftrightarrow \sqrt{ab} - \text{целое})$

    Прием имеет заголовок "второйтерм". Утверждение о целочисленности произведения является условием задачи на описание и содержит неизвестные. Уровень срабатывания равен 3.

    $\forall_{abcdex}(a + b = c\ \&\ c = 2d\ \&\ e = d^2 - ab\ \&\ x - \text{целое} \rightarrow \sqrt{(x + a)(x + b)} - \text{целое} \leftrightarrow \exists_{mn}(e = mn\ \&\ x = (m + n)/2 - d))$

    Утверждение является условием задачи на описание. Выражение x содержит неизвестные, выражения a, b суть целочисленные константы. Первые три антецедента выделены указателем "программа", последний - обрабатывается проверочным оператором. Указатель "или" обеспечивает развертку заменяющего квантора существования в дизъюнкцию . Количество натуральных делителей числа e не должно превышать 19. Уровень срабатывания равен 3.

    $\forall_{abcnp}(a - \text{целое}\ \&\ p|b\ \&\ p|c\ \&\ \neg(p^2|c)\ \&\ n - \text{целое}\ \&\ 0 \leq n\ \&\ \sqrt{ab^n + c} - \text{целое} \rightarrow n = 0\ \vee\ n = 1)$

    Прием имеет заголовок "выводусловия". Последний антецедент идентифицируется с условием задачи на описание. Показатель степени n содержит неизвестные, выражения b, c суть целочисленные константы. Антецеденты со второго по четвертый выделены указателем "программа". При этом устанавливается существование такого простого числа p, делящего b и c, что его квадрат не делит c. Компилятор воспринимает фильтр "простое(p)" как указание на использование процедуры, перечисляющей простые делители p числа b. Оставшиеся антецеденты обрабатываются проверочными операторами. Выводимое утверждение снабжается примечанием "разборслучаев". Уровень срабатывания равен 3.

    $\forall_{abc}(a - \text{целое}\ \&\ b - \text{целое}\ \&\ c - \text{целое}\ \&\ d = c - b^2\ \&\ 0 < d\ \&\ \sqrt{a^2 + 2ab + c} - \text{целое} \rightarrow 2|a + b| + 1 \leq d)$

Прием имеет заголовок "выводусловия". Последний антецедент идентифицируется с условием задачи на описание. Хотя бы одно из выражений a, b содержит неизвестные. Четвертый антецедент выделен указателем "идентификатор", его правая часть обрабатывается нормализаторами общей стандартизации. Оставшиеся антецеденты обрабатываются проверочными операторами. Уровень срабатывания равен 4.

(b) Условие целочисленности логарифма.

$\forall_{mn}(\log_m n$ – целое $\leftrightarrow \exists_k(k$ – целое & $n = m^k))$

Утверждение о целочисленности логарифма является условием задачи на описание. Выражение n содержит неизвестные, выражение m - не содержит. Преобразованное условие снабжается комментарием "серия". Уровень срабатывания равен 4.

$\forall_{abcmnkp}(k$ – целое & m – целое & n – целое & p – целое & a – целое & $\neg(a = 0)$ & b – целое & $\neg(b = 0)$ & c – целое & $m\log_k n + p\log_k(a + b\sqrt{c})$ – целое $\rightarrow \sqrt{c}$ – целое$)$

Прием имеет заголовок "выводусловия". Последний антецедент идентифицируется с условием задачи на описание, остальные - обрабатываются проверочными операторами. Выражение c содержит неизвестные. Целочисленность выражения $\sqrt{c}$ не должна усматриваться из контекста с помощью проверочного оператора. Уровень срабатывания равен 3.

(c) Сложение либо вычитание целочисленных выражений с неизвестными.

$\forall_{abcd}((a + b)/c$ – целое & $(d - b)/c$ – целое $\rightarrow (a + d)/c$ – целое$)$

$\forall_{abcd}((a + b)/c$ – целое & $(b + d)/c$ – целое $\rightarrow (a - d)/c$ – целое$)$

Прием имеет заголовок "выводусловия". Антецеденты идентифицируются с условиями задачи на описание. Выражение c содержит неизвестные, выражения a, d - не содержат. Целью вывода является исключение неизвестных в числителе. Уровень срабатывания равен 5.

(d) Умножение целочисленных выражений с неизвестными.

$\forall_{abcd}(d = (a^2 - b^2)/c^2$ & $(a - b)/c$ – целое & $(a + b)/c$ – целое $\rightarrow d$ – целое$)$

Прием имеет заголовок "выводусловия". Два последних антецедента идентифицируются с условиями задачи на описание, первый - выделен указателем "идентификатор". Его правая часть обрабатывается нормализаторами общей стандартизации и нормализатором раскрывания скобок "стандплюс". Хотя бы одно из выражений a, b имеет своим множителем радикал с неизвестными. Уровень срабатывания равен 6.

(e) Ввод вспомогательной целочисленной неизвестной.

$\forall_{nx}(nx$ – целое $\leftrightarrow \exists_m(m$ – целое & $x = m/n))$

Заменяемое утверждение является условием задачи на описание. Переменная x есть неизвестная, выражение n не содержит неизвестных. Преобразованное условие сопровождается комментарием "серия". После применения приема будет введена вспомогательная задача, где m будет играть роль дополнительной неизвестной. Уровень срабатывания равен 5.

$\forall_a(a$ – целое $\leftrightarrow \exists_n(n$ – целое & $a = n))$

Заменяемое утверждение является условием задачи на описание. Выражение a содержит неизвестные, но отлично от неизвестной. Преобразованное утверждение сопровождается комментарием "серия". Уровень срабатывания равен 9.

(f) Вывод неравенства для модулей числителя и знаменателя из условия целочисленности.

$\forall_{ab}(\neg(a = 0)\ \&\ a/b - \text{целое} \to |b| \leq |a|)$

Прием имеет заголовок "выводусловия". Второй антецедент идентифицируется с условием задачи на описание, первый - обрабатывается проверочным оператором. Выражение b содержит неизвестные, выражение a - не содержит. Уровень срабатывания равен 6.

(g) Получение параметрического описания для условия целочисленноси выражения с неизвестной.

$\forall_{abnxP}((a = n) = (x = b\ \&\ P(n)) \to a - \text{целое} \leftrightarrow \exists_n(n - \text{целое}\ \&\ x = b\ \&\ P(n)))$

Утверждение "a – целое" является условием задачи на описание. Выражение a содержит неизвестную x, отлично от переменной и не содержит других неизвестных. Антецедент выделен указателем "идентификатор". Его левая часть разрешается относительно x с помощью вспомогательной задачи на описание. В качестве дополнительной посылки этой задаче передается утверждение "n – целое", где n - некоторая новая переменная. Переменная P - функциональная. Задача не должна иметь еще одного условия целочисленности. Преобразованное условие сопровождается комментарием "серия". Уровень срабатывания равен 4.

16. Усмотрение существования значения неизвестной.

$\forall_{an}(b - \text{число} \to \exists_x(x - \text{целое}\ \&\ \forall_i(i \in \{1, \ldots, n\} \to \neg(x = a(i)))\ \&\ b < x))$

Прием имеет заголовок "связка". Переменная x идентифицируется с несущественной неизвестной задачи на описание. Квантор общности выделен указателем "развертка", т.е. все содержащие x условия рассматриваемой задачи идентифицируются с утверждениями "$x-$целое$, \neg(x = a(1)), \ldots, \neg(x = a(n)), b < x$". Переменная a - функциональная. Допускаются нестрогое неравенство и изменение знака неравенства на противоположный. Выражения $a(1), \ldots, a(n), b$ не содержат x. Уровень срабатывания равен 3.

$\forall_a(a - \text{число} \to \exists_x(x - \text{число}\ \&\ \neg(ax - \text{целое})) \leftrightarrow \neg(a = 0))$

Прием имеет заголовок "связка". Уровень срабатывания равен 2.

$\forall_{nA}(n - \text{целое} \to \exists_{ij}(i - \text{целое}\ \&\ j - \text{целое}\ \&\ 1 \leq j - i\ \&\ 1 \leq i\ \&\ j \leq n\ \&\ i \in A\ \&\ j \in A) \leftrightarrow 2 \leq \text{card}(A \cap \{1, \ldots, n\}))$

$\forall_{nA}(n - \text{целое} \to \exists_{ij}(i - \text{натуральное}\ \&\ j - \text{целое}\ \&\ 1 \leq j - i\ \&\ j \leq n\ \&\ i \in A\ \&\ j \in A) \leftrightarrow 2 \leq \text{card}(A \cap \{1, \ldots, n\}))$

Прием имеет заголовок "связка". Переменные i, j суть неизвестные задачи на описание, причем все содержащие эти неизвестные условия идентифицируются с подкванторными утверждениями. Уровень срабатывания равен 5.

$\forall_{an}(\exists_x(\forall_i(i \in \{1, \ldots, n\} \to \neg(x = a(i)))\ \&\ x - \text{целое}))$

$\forall_{an}(\exists_x(\forall_i(i \in \{1, \ldots, n\} \to \neg(x = a(i)))\ \&\ x - \text{натуральное}))$

Квантор общности выделен указателем "развертка". Уровень срабатывания равен 2.

17. Попытка усмотрения невыполнимости уравнения по заданному простому модулю.

    $\forall_{kmnpqxy}(x-$целое & $y-$целое & $p|n$ & простое(p) & $\neg(\exists_i(i \in \{0,\ldots,p-1\}$ & $(mi^k-q)(\mathrm{mod}p)=0)) \rightarrow \neg(mx^k+ny=q))$

    Прием имеет заголовок "второйтерм". Уравнение идентифицируется с условием задачи на описание. Выражения x, y содержат неизвестные, выражения m, n, q суть целочисленные константы, k - натуральная константа, не превосходящая 6. Первые два антецедента обрабатываются проверочными операторами, остальные - выделены указателями "программа". Здесь перечисляются всевозможные простые делители p числа n, и для всех $i = 0, \ldots, p-1$ вычисляются значения по модулю p выражения $mi^k - q$. Если при каком-либо p эти значения оказываются ненулевыми, делается вывод о неразрешимости уравнения. Уровень срабатывания равен 5.

18. Нормализатор решения целочисленных уравнений "уравнцелое". Нормализатор используется редко и имеет всего два приема - для решения уравнений вида $a + x = b$ и для решения линейного целочисленного уравнения с двумя неизвестными:

    $\forall_{abcpqxy}(x-$ целое & $y-$ целое & $a \leq b$ & $b = ap+q \rightarrow ax+by=c \leftrightarrow \exists_z(x = z-py$ & $z-$ целое & $az+qy=c))$

Усмотрение параметрического описания неизвестной

$\forall fz(\exists_n(n-$ целое & $z=f(n)) \rightarrow \emptyset)$

Прием имеет заголовок "замечание". Антецедент идентифицируется с условием задачи на описание, причем переменная z - с неизвестной задачи. Выражение $f(n)$ неизвестных не содержит, переменная f - функциональная. Если рассматриваемое условие не имело комментария "серия", то оно сопровождается данным комментарием, указывающим, что условие представляет собой параметрическое описание неизвестной. Уровень срабатывания равен 1.

Выдача ответа задачи на описание, имеющей целочисленную неизвестную

Используются следующие шаблоны ответов:

1. "явное(x14 набор(целое(x14)не(четное(x14))меньшеилиравно(x14 x1))пустоеслово пустоеслово)". Все содержащие неизвестную n условия задачи содержаться в списке: целое$(n), \neg(n-\mathrm{even}), n \leq a$.

2. "явное(x14 набор(целое(x14)четное(x14)меньшеилиравно(x14 x1)) пустоеслово пустоеслово)". Все содержащие неизвестную n условия задачи содержатся в списке: целое$(n), n-\mathrm{even}, n \leq a$.

3. "явное(x14 набор(целое(x14)меньшеилиравно(x1 x14)меньшеилиравно(x14 x2)) набор(не(равно(x14 x1))взаимнопросты(x14 x3))пустоеслово)". Все содержащие неизвестную n условия задачи содержатся в списке: целое$(n), a \leq n, n \leq b$, к которому разрешается добавлять произвольное число утверждений вида $\neg(n = m)$, взаимнопросты(m, n).

Определение максимума либо минимума по множеству целых точек: локальное условие

$\forall_{fmnpyzA}(\mathrm{Max}(f,\{m,\ldots,n\},y,z)\ \&\ f=\lambda_x(p(x),A(x))\ \&\ \forall_i(i\in\{m,\ldots,n\}\to A(i))\to y\subseteq \mathrm{set}_i(i\in\{m,\ldots,n\}\ \&\ (i=m\ \vee\ 0\leq p(i)-p(i-1))\ \&\ (i=n\ \vee\ 0\leq p(i)-p(i+1))))$

Прием имеет заголовок "вывод". Первые два антецедента идентифицируются с посылками задачи на исследование. Выражение y содержит неизвестные, выражения m,n - не содержат. Таким образом, стоит задача определения множества y точек целочисленного отрезка $\{m,\ldots,n\}$, в которых явно заданная описателем "отображение" функция f принимает наибольшее значение z. Третий антецедент обрабатывается вспомогательной задачей на доказательство. Он проверяет, что рассматриваемый целочисленный отрезок входит в область определения f. Выводимое утверждение означает, что точки i искомого подмножества y должны удовлетворять условию локального экстремума: значения функции в точках $i\pm 1$ не должны превосходить $f(i)$. Утверждение под описателем "класс" явно разрешается с помощью вспомогательной задачи на описание относительно i. Уровень срабатывания равен 3.

Аналогичный прием имеется для поиска минимума.

Усмотрение множества целых чисел

$\forall_z(z - \text{целое}) = \mathbf{Z}$

Уровень срабатывания равен 1.

Проверочный оператор "усмцелое"

Оператор усматривает истинность утверждения "целое(x)". Кроме приема непосредственного извлечения данного утверждения из списка посылок, он имеет следующие приемы:

1. Отбрасывание минуса.

   $\forall_n(n - \text{целое} \to -n - \text{целое})$

   Антецедент обрабатывается проверочным оператором.

2. Использование посылки вида "натуральное(n)".

   $\forall_n(n - \text{натуральное} \to n - \text{целое})$

   $\forall_n((-n) - \text{натуральное} \to n - \text{целое})$

   Антецедент идентифицируется с посылкой. Выражение n неконстантное.

3. Сумма.

   $\forall_{ab}(a - \text{целое}\ \&\ b - \text{целое} \to (a+b) - \text{целое})$

   Указатель "дистрибразвертка(фикс(0 1))" обеспечивает одновременную обработку всех слагаемых суммы. Для обработки используется проверочный оператор.

4. Произведение.

   $\forall_{ab}(a - \text{целое}\ \&\ b - \text{целое} \to ab - \text{целое})$

   Аналогично предыдущему.

5. Степень.

   $\forall_{ab}(a$ – целое & $0 \le a$ & b – целое $\rightarrow b^a$ – целое$)$

   $\forall_n(n$ – целое $\rightarrow (-1)^n$ – целое$)$

   $\forall_{ab}(a$ – натуральное & b – целое $\rightarrow b^a)$

   Антецеденты здесь и в следующих трех приемах обрабатываются проверочными операторами.

6. Модуль.

   $\forall_a(a$ – целое $\rightarrow |a|$ – целое$)$

7. Сигнум.

   $\forall_a(\mathrm{sg}a$ – целое$)$

8. Мощность конечного множества.

   $\forall_a($конечное$(a) \rightarrow \mathrm{card}a$ – целое$)$

9. Принадлежность конечному отрезку целых чисел.

   $\forall_{amn}(a \in \{n, \ldots, m\} \rightarrow a$ – целое$)$

   $\forall_{amn}(a \in \{n, \ldots, m\} \backslash b \rightarrow a$ – целое$)$

   В обоих случаях антецедент идентифицируется с посылкой.

   $\forall_{amnA}(a \in A$ & $A \subseteq \{m, \ldots, n\}$ & m – целое & n – целое $\rightarrow a$ – целое$)$

   Первые два антецедента идентифицируются с посылками, последние два - обрабатываются проверочными операторами.

10. Целая часть.

    $\forall_a([a]$ – целое$)$

11. Наибольший общий делитель.

    $\forall_{mn}(m$ – целое & n – целое $\rightarrow$ нод(m, n) – целое$)$

    Антецеденты здесь и в следующих двух приемах обрабатываются проверочными операторами.

12. Наименьшее общее кратное.

    $\forall_{mn}(m$ – целое & n – целое $\rightarrow$ нок(m, n) – целое$)$

13. Дробь.

    $\forall_{an}(\neg(a = 0)$ & $a|n$ & n – целое $\rightarrow n/a$ – целое$)$

14. Усмотрение из принадлежности конечному списку целых чисел.

    $\forall_{an}(n \in \{; a\}$ & $\{; a\} \subseteq \mathbf{Z} \rightarrow n$ – целое$)$

    Первый антецедент идентифицируется с посылкой, второй - обрабатывается проверочным оператором.

15. Длина набора.

    $\forall_a(l(a) - \text{целое})$

    $\forall_n(n = l(a) \to n - \text{целое})$

    Антецедент идентифицируется с посылкой.

16. Усмотрение из равенства в посылках.

    $\forall_{mn}(m = n \to m - \text{целое})$

    Антецедент идентифицируется с посылкой, причем n - целочисленная константа.

17. Степень многочлена.

    $\forall_a(\deg(a) - \text{целое})$

18. Использование кванторной посылки.

    $\forall_{afP}(P(a) \mathbin{\&} \forall_x(P(x) \to f(x) - \text{натуральное}) \to f(a) - \text{целое})$

    Второй антецедент идентифицируется с посылкой. Переменная P - функциональная, f - обычная. Первый антецедент обрабатывается вспомогательной задачей на доказательство.

19. Использование специальных посылок.

    $\forall_{An}(\text{вертикнули}(A, n) \to n - \text{целое})$

    Утверждение "вертикнули(A n)" используется в приемах, диагонализирующих квадратную матрицу методом Гаусса. Оно означает, что матрица A имеет нули под главной диагональю в столбцах, номера которых меньше n.

    $\forall_{amnA}(A = \text{Матрица}(a, m, n) \to m - \text{целое})$

    $\forall_{amnA}(A = \text{Матрица}(a, m, n) \to n - \text{целое})$

    Выражение "Матрица(a m n)" обозначает прямоугольную матрицу, имеющую m строк и n столбцов, порожденную набором a. Оно используется в вычислительных приемах ГЕНОЛОГа. Строки матрицы формируются как последовательные отрезки набора a.

    $\forall_{abcmn}(\text{Pmax}(\lambda_i(a(i), i \in \{m, \ldots, n\}), b, c) \to b - \text{целое})$

    Утверждение $\text{Pmax}(f, b, c)$ означает, что b есть точка, в которой функция f принимает наибольшее значение c на своей области определения. Переменная a - функциональная.

    $\forall_{kmnp}(m(\text{mod}) + p = 0 \mathbin{\&} n|(k - p) \to (m + k)/n - \text{целое})$

    Первый антецедент берется в посылках. Переменные k, p идентифицируются с целочисленными константами, переменная n - с натуральной константой. Второй антецедент выделен указателем "программа".

20. Целочисленная область значений функции.

    $\forall_{fi}(\text{Val}(f) \subseteq \mathbf{Z} \to f(i) - \text{целое})$

    Переменная f- обычная. Антецедент обрабатывается проверочным оператором.

21. Элемент набора целых чисел.

    $\forall_{ani}(i \in \{1, \ldots, n\}$ & кортеж$(a, n, \mathbf{Z}) \to a(i)$ – целое)

    Второй антецедент берется в посылках, первый - обрабатывается проверочным оператором. Вместо множества целых чисел во втором антецеденте может рассматриваться множество натуральных чисел.

22. Простое число.

    $\forall_n$(простое$(n) \to n$ – целое)

    Антецедент берется в посылках.

23. Числитель и знаменатель.

    $\forall_a$(числитель(a) – целое)

    $\forall_a$(знаменатель(a) – целое)

24. Конечная сумма.

    $\forall_{fP}(f(i)$ – целое $\to \sum_{i,P(i)} f(i)$ – целое)

    Антецедент обрабатывается проверочным оператором, причем вводится дополнительная посылка $P(i)$. Переменные f, P - функциональные.

25. Порядок элемента группы.

    $\forall_{aA}$(порядокэлемента(a, A) – число $\to$ порядокэлемента(a, A) – целое)

    Антецедент обрабатывается проверочным оператором. Фактически при этом проверяется, что порядок элемента конечный.

26. Наименьший элемент подмножества натуральных чисел.

    $\forall_P(\inf(\mathrm{set}_x(P(x)$ & x – натуральное)) – целое)

    Переменная P - функциональная.

27. Четность перестановки.

    $\forall_a$(четность(a) – целое)

28. Число переменных функции.

    $\forall_f$(числперем(f) – целое)

Проверочный оператор "усмнецелое"

Оператор усматривает истинность утверждения "не(целое(n))".

1. Отбрасывание минуса.

   $\forall_n(\neg(n$ – целое$) \to \neg((-n)$ – целое))

   Антецедент обрабатывается проверочным оператором.

2. Сумма дробей с четным и нечетным знаменателем.

   $\forall_{ab}$(знаменатель(b) – even & $\neg$(знаменатель(a) – even) $\to \neg((a + b)$ – целое))

   Антецеденты обрабатываются проверочными операторами.

3. Числовая дробь.

   $\forall_{ab}(a < b \rightarrow \neg(a/b - \text{целое}))$

   $\forall_{abcd}(a = bc + d \mathbin{\&} \neg(d = 0) \rightarrow \neg(a/b - \text{целое}))$

   Переменные a, b идентифицируются с натуральными константами. Антецеденты выделены указателем "программа".

4. Все слагаемые числителя, кроме одного, а также знаменатель делятся на некоторое простое число.

   $\forall_{abcn}(n - \text{целое} \mathbin{\&} p|a \mathbin{\&} p|c \mathbin{\&} \neg(p|b) \rightarrow \neg((an + b)/c - \text{целое}))$

   Переменные a, b, c идентифицируются с целочисленными константами. Первый антецедент обрабатывается проверочным оператором, остальные - выделены указателем "программа". Второй антецедент перечисляет всевозможные простые делители p числа a. Компилятору указывает на это фильтр "простое(p)". Затем проверяется, что c делится на p, а b - не делится.

5. Дробь, у которой модуль знаменателя больше модуля числителя.

   $\forall_{mn}(a \leq |n| - |m| \mathbin{\&} 0 < a \rightarrow \neg(m/n - \text{целое}))$

   Первый антецедент обрабатывается синтезатором "нижняяоценка", второй - проверочным оператором.

Нормализатор общей стандартизации "нормцелое"

Нормализатор предпринимает общую стандартизацию утверждений вида "a−целое". Он содержит следующие приемы, дублирующие приведенные выше приемы сканирования задачи:

1. Отбрасывание минуса в условии целочисленности.

2. Отбрасывание заведомо целочисленного слагаемого.

3. Усмотрение целочисленного выражения с помощью оператора "усмцелое".

4. Целочисленность дроби с константным числителем, знаменатель которой стремится к бесконечности, эквивалентна равенству нулю числителя.

13.2 Приемы символа "натуральное"

Утверждение "натуральное(a)", прорисовываемое формульным редактором в виде "a – натуральное", означает, что a есть натуральное число. Символ характеризуется справочниками "арность", "предикатныйсимвол", "род", "родобъекта". Он задает один из основных типов объектов, причем надтипами являются типы "целое", "рациональное", "число", "комплексное".

Усмотрение натурального числа

Для усмотрения истинности утверждений "$f(\ldots)$ – натуральное" с помощью справочника "тип", определяющего натуральный тип значений операции либо константы f, служит прием с заголовком "родобъекта" и теоремой "родобъекта(натуральное)". Уровень его срабатывания равен 0.

Усмотрение множества натуральных чисел

$\mathrm{set}_z(z -$ натуральное$) = \mathbf{N}$

Уровень срабатывания равен 1.

Общая стандартизация утверждений

1. Исключение минуса.

   $\forall_n((-n) -$ натуральное $\leftrightarrow n -$ целое $\& \; n+1 \leq 0)$

   Уровень срабатывания равен 1.

2. Отбрасывание слагаемого.

   $\forall_{mn}(m -$ целое $\rightarrow (m+n) -$ натуральное $\leftrightarrow n -$ целое $\& \; 0 \leq m+n-1)$

   Антецедент обрабатывается проверочным оператором.

3. Исключение дроби.

   $\forall_{mn}(n -$ натуральное $\& \; m -$ натуральное $\rightarrow (m/n) -$ натуральное $\leftrightarrow n|m)$

   Антецеденты обрабатываются проверочными операторами. Уровень срабатывания равен 1.

4. Исключение квантора.

   $\exists_n(n -$ натуральное$)$

   Прием имеет заголовок "второйтерм". Уровень срабатывания равен 0.

   $\forall_{an}(\exists_x(x -$ натуральное $\& \; \forall_i(i \in \{1, \dots, n\} \rightarrow \neg(x = a(i)))))$

   Заголовок тот же. Квантор общности выделен указателем "равзертка" и идентифицируется с конъюнкцией утверждений $\neg(x = a(1)), \dots, \neg(x = a(n))$. Переменная a - функциональная. Уровень срабатывания равен 1.

   $\forall_{mn}(m -$ натуральное $\& \; n -$ целое $\rightarrow \forall_i(i -$ натуральное $\& \; i \leq m \rightarrow i < n) \leftrightarrow m < n)$

   Антецеденты обрабатываются проверочными операторами. Уровень срабатывания равен 1.

   $\forall_n(\neg(\forall_i(i -$ натуральное $\rightarrow i < n)))$

   Прием имеет заголовок "второйтерм". Уровень срабатывания равен 1.

   $\forall_{fAB}(f(x) -$ натуральное $\rightarrow \forall_{xyz}(f(x) = yz \; \& \; y -$ натуральное $\& \; z -$ целое $\& \; A(x,z) \rightarrow B(x,z)) \leftrightarrow \forall_{xy}(y|f(x) \; \& \; y -$ натуральное $\& \; A(x,y) \rightarrow B(x,y)))$

   Прием уменьшает число связываемых кванторной импликацией переменных. Переменные f, A, B функциональные. Антецедент обрабатывается проверочным оператором, причем вводится дополнительная посылка $A(x,z)$. Указатель "кортежпеременных(x23)" разрешает идентифицировать x со списком переменных произвольной длины. Уровень срабатывания равен 2.

5. Усмотрение истинности неравенств и отрицаний равенств нулю для натуральной переменной. Все приемы этого раздела имеют заголовок "второйтерм" и применяются к подутверждению посылки задачи.

   $\forall_n(n -$ натуральное $\rightarrow 0 \leq n-1)$

Антецедент берется в контексте. Уровень срабатывания равен 1.

$\forall_{abn}(0 \leq a \ \& \ 0 \leq b \ \& \ n -$ натуральное $\rightarrow 0 \leq an + b)$

$\forall_{abn}(0 < a \ \& \ 0 \leq b \ \& \ n -$ натуральное $\rightarrow \neg(an + b = 0))$

$\forall_{abn}(0 < a \ \& \ 0 \leq b \ \& \ n -$ натуральное $\rightarrow 0 < an + b)$

Выражения a, b константные. Последний антецедент берется в контексте, первые два - обрабатываются проверочными операторами. Уровень срабатывания равен 1.

6. Замена условия натуральности на условие целочисленности в описателе "отображение".

   $\forall_{ab}(\lambda_n(a(n), n -$ натуральное $\& \ n \leq b) = \lambda_n(a(n), n -$ целое $\& \ 1 \leq n \ \& \ n \leq b))$

   Предполагается, что описатель является непосредственным операндом одной из одноместных операций "суммавсех", "объединениевсех", "пересечениевсех", "произведениевсех". Прием выполняет стандартизацию задания области определения, на которую рассчитаны приемы, связанные с этими операциями. Переменная a функциональная. Уровень срабатывания равен 1.

   $\forall_{fP}(\lambda_{xy}(f(x, y), x -$ натуральное $\& \ P(x, y)) = \lambda_{xy}(f(x, y), x -$ целое $\& \ 1 \leq x \ \& \ P(x, y)))$

   Аналогично предыдущему. Переменные f, P функциональные. Допускается идентификация y со списком переменных произвольной (в том числе нулевой) длины.

7. Сдвиг натуральной параметризации.

   $\forall_A(\exists_n(n -$ натуральное $\& \ \neg(n = 1) \ \& \ A(n)) \leftrightarrow \exists_n(n -$ натуральное $\& \ A(n + 1)))$

   Переменная A функциональная. Уровень срабатывания равен 2.

8. Исключение избыточных ограничений на натуральный параметр.

   $\forall_{fA}(A(n) \rightarrow \mathrm{set}_x(\exists_n(n -$ натуральное $\& \ A(n) \ \& \ x = f(n))) = \mathrm{set}_x(\exists_n(n -$ натуральное $\& \ x = f(n))))$

   Переменные f, A функциональные. Антецедент обрабатывается вспомогательной задачей на доказательство, снабженной дополнительной посылкой "n – натуральное". Уровень срабатывания равен 1.

   $\forall_A(\mathrm{set}_x(x -$ натуральное $\& \ 1 \leq x \ \& \ A(x)) = \mathrm{set}_x(x -$ натуральное $\& \ A(x)))$

   Переменная A функциональная. Уровень срабатывания равен 1.

Уравнения в натуральных числах

1. Усмотрение промежутка значений натуральной неизвестной.

   $\forall_{abcdmn}(n -$натуральное $\& \ 0 < a \ \& \ d = [(c - m)/a] \ \& \ m \leq b \ \& \ an + b = c \rightarrow n \leq d)$

   Прием имеет заголовок "выводусловия". Первый и последний антецеденты идентифицируются с условиями задачи на описание. Переменная n идентифицируется с неизвестной этой задачи. Выражения a, c суть целочисленные константы. Второй антецедент обрабатывается проверочным оператором. Четвертый антецедент использует синтезатор "нижняяоценка" для нахождения нижней оценки m выражения b. Третий антецедент, выделенный указателем "идентификатор", вычисляет целую часть дроби, используя нормализатор "нормцелаячасть". Результат d представляет собой целочисленную константу. Задача не должна иметь условия вида $n \leq k$, где k - десятичная константа, не превосходящая d. Уровень срабатывания равен 8.

2. Равенство единице произведения двух натуральных множителей.

   $\forall_{ab}(a - \text{натуральное} \ \& \ b - \text{натуральное} \rightarrow ab = 1 \leftrightarrow a = 1 \ \& \ b = 1)$

   Прием имеет заголовок "второйтерм". Равенство идентифицируется с условием задачи на описание. Проверяется, что эта задача имеет условие вида "натуральное(...)". Антецеденты обрабатываются проверочными операторами. Уровень срабатывания равен 2.

3. Замена условия натуральности неизвестного целочисленного выражения на неравенство.

   $\forall_n(n - \text{целое} \rightarrow n - \text{натуральное} \leftrightarrow 0 \leq n - 1)$

   Утверждение входит в условие задачи на описание. Выражение n содержит неизвестные и отлично от переменной. Антецедент обрабатывается проверочным оператором. Уровень срабатывания равен 4.

   $\forall_n(n - \text{натуральное} \leftrightarrow n - \text{целое} \ \& \ 0 \leq n - 1)$

   Аналогично предыдущему, но дополнительно требуется, чтобы n не имело вида "значение(...)". Уровень срабатывания равен 5.

   $\forall_n(n - \text{целое} \rightarrow \neg(n - \text{натуральное}) \leftrightarrow n \leq 0)$

   Отрицание требования натуральности является условием задачи на описание, причем имеет место этап редактирования ответа. Антецедент идентифицируется в контексте замены. Уровень срабатывания равен 1.

4. Усмотрение существования значения неизвестной. Приемы этого раздела имеют заголовок "связка":

   $\forall_a(a{-}\text{число} \ \& \ 0 < a \rightarrow \exists_n(n{-}\text{натуральное} \ \& \ \neg(a^{1/n}{-}\text{натуральное})) \leftrightarrow \neg(a = 1))$

   Переменная n идентифицируется с неизвестной задачи на описание, причем все содержащие ее условия суть: "n – натуральное, $\neg(a^{1/n}$ – натуральное)". Антецеденты обрабатываются проверочными операторами. Уровень срабатывания равен 1.

   $\forall_{kn}(n - \text{натуральное} \ \& \ k - \text{натуральное} \rightarrow \exists_m(m - \text{натуральное} \ \& \ 0 < n - m \ \& \ \neg(m = k)) \leftrightarrow 3 \leq n \ \vee \ n = 2 \ \& \ \neg(k = 1))$

   Переменная m идентифицируется с неизвестной задачи на описание, причем все содержащие ее условия суть: "m – натуральное, $0 < n - m$, $\neg(m = k)$". Антецеденты обрабатываются проверочными операторами. Уровень срабатывания равен 1.

   $\forall_{an}(\exists_x(x - \text{натуральное} \ \& \ \forall_i(i \in \{1, \ldots, n\} \rightarrow \neg(x = a(i)))))$

   Переменная x идентифицируется с неизвестной задачи на описание, причем все содержащие ее условия суть: "x – натуральное, $\neg(x = a(1)), \ldots, \neg(x = a(n))$". Переменная a - функциональная. Уровень срабатывания равен 3.

5. Подбор примера.

   $\forall_a(a = 1 \to a -$ натуральное$)$

   Прием имеет заголовок "подборзначений". Консеквент идентифицируется с условием задачи на описание, имеющей цель "пример", причем переменная a - с неизвестной задачи. В другие условия a не входит. Антецедент выделен указателем "подборзначений" - он замещает рассматриваемое условие при попытке подбора значения. Указатель "новый" инициирует попытки применения приема при каждом возобновлении сканирования "теневой" области задачи. Это нужно для контроля моментов исключения всех прочих условий, содержащих a. Уровень срабатывания равен 0.

   $\forall_{ab}(0 < a \ \& \ 0 \leq b - 1 \ \& \ a < b \to \forall_n(n -$ натуральное $\to \neg(a = b^n)))$

   Консеквент идентифицируется с кванторной импликацией - условием задачи на описание. Выражение a содержит неизвестные. Первые два антецедента обрабатываются проверочным оператором, третий - выделен указателем "подборзначений". Уровень срабатывания равен 3.

6. Использование синтезатора "верхняяоценка" для усмотрения конечного промежутка значений натурального выражения с неизвестными.

   $\forall_{abcx}(x -$ натуральное $\& \ ax + b = 0 \ \& \ -b/x \leq c \to a \leq c)$

   Прием имеет заголовок "выводусловия". Первые два антецедента идентифицируются с условиями задачи на описание, переменная x - неизвестная задачи. Выражения a, b не содержат x. Первое из этих выражений содержит неизвестные, причем усматривается, что оно принимает натуральные значения. Третий антецедент обрабатывается синтезатором "верхняяоценка", определяющим верхнюю оценку c. Предварительно дробь $-b/x$ обрабатывается нормализатором раскрывания скобок "стандплюс". Уровень срабатывания равен 6.

7. Усмотрение параметрического описания.

   $\forall_n(n -$ целое $\& \ 0 < n \to \forall_k(k -$ натуральное $\to \neg(n = 2k)) \leftrightarrow \exists_m(m -$ натуральное $\& \ n = 2m + 1))$

   Кванторная импликация является условием задачи на описание. Переменная n идентифицируется с неизвестной задачи. Антецеденты обрабатываются проверочными операторами. Преобразованное условие сопровождается комментарием "серия". Уровень срабатывания равен 2.

Доказательство индукцией по натуральному параметру

1. Простейшая схема индукции по натуральному параметру.

   $\forall_{mg}(m -$ натуральное $\& \ g(m) \ \& \ \forall_n(n -$ натуральное $\& \ g(n) \ \& \ 0 \leq n - m \to g(n+1)) \to \forall_n(n -$ натуральное $\& \ 0 \leq n - m \to g(n)))$

   Прием имеет заголовок "свертка". Это означает, что идентификация кванторной импликации $\forall_n(n -$ натуральное $\& \ 0 \leq n - m \to g(n))$ происходит одновременно в условии и посылках задачи на доказательство. Утверждение $g(n)$ идентифицируется с условием задачи, а антецеденты "$n -$натуральное, $0 \leq n - m$" - с некоторыми ее посылками. При идентификации предполагается, что переменная g - функциональная. Указатель "подстановка(фикс(0 4)x13 1)"разрешает отсутствовать в посылках утверждению $0 \leq n-m$, и тогда m берется равным 1. В любом случае m должно представлять собой натуральную константу. Очевидно, что для привязки приема остается лишь логический символ "натуральное", который и выбирается компилятором. Таким образом, инициализация попытки применения приема происходит при усмотрении посылки задачи на доказетельство, имеющей вид "$m-$натуральное". Первый антецедент обрабатывается проверочным оператором. Второй и третий антецеденты (соответственно, базис индукции и шаг индукции) выделены указателем "следствие". Они обрабатываются вспомогательными задачами на доказательство. При доказательстве шага индукции могут понадобиться специальные приемы, связывающие текущее утверждение $g(n+1)$ с индуктивным предположением $g(n)$. Для их подключения служит комментарий (целое n) к индуктивному предположению, создаваемый указателем "комментарий(фикс(3 4)целое переменная(x14))". Кроме того, вспомогательной задаче шага индукции передается комментарий (натуральное n), блокирующий повторную попытку индукции по n. Блокировка повторной попытки применения приема в текущей задаче выполняется комментарием (натуральное доказать n). Уровень срабатывания равен 7.

2. Индукция по длине набора. Если натуральный параметр характеризует некоторый нечисловой объект, то приходиться создавать приемы доказательства по индукции, в которых варьируются как параметр, так и сам объект. Простейший пример - доказательство индукцией по длине набора:

   $\forall_{mPQ}(m -$ натуральное & $\forall_f(f -$ функция & $\forall_i(i \in \{1,\ldots,m\} \to P(f(i))) \to Q(f,m))$ & $\forall_{fn}(n-$натуральное & $f-$функция & $0 \leq n-m$ & $\forall_i(i \in \{1,\ldots,n+1\} \to P(f(i)))$ & $Q(f,n) \to Q(f,n+1)) \to \forall_{fn}(n -$ натуральное & $f -$ функция & $0 \leq n-m$ & $\forall_i(i \in \{1,\ldots,n\} \to P(f(i))) \to Q(f,n)))$

   Как и выше, заголовок приема - "свертка". Условие задачи на доказательство идентифицируется с утверждением $Q(f,n)$, некоторые из ее посылок - с утверждениями "n – натуральное, f – функция, $0 \leq n-m$, $\forall_i(i \in \{1,\ldots,n\} \to P(f(i)))$". Переменные P,Q функциональные, переменная f обычная. Указатель "консеквент(фикс(0 7 5))" обеспечивает склейку всех имеющихся в посылках кванторных импликаций вида $\forall_i(i \in \{1,\ldots,n\} \to A(i))$ в одну общую импликацию, идентифицируемую с $\forall_i(i \in \{1,\ldots,n\} \to P(f(i)))$. Консеквент $P(f(i))$ идентифицируется с конъюнкцией соответствующих $A(i)$. Указатель "новаргумент(x40 x6 фикс)" проверяет, что внутри этой конъюнкции переменная f встречается только в виде $f(i)$. Указатель "подстановка(фикс(0 6)x13 1)" допускает случай отсутствия посылки $0 \leq n-m$, и тогда m идентифицируется с 1. Первый антецедент обрабатывается проверочным оператором, второй и третий (базис и шаг индукции) - обрабатываются вспомогательными задачами на доказательство. Используются те же комментарии (натуральное n), (целое n), что и выше. Уровень срабатывания равен 5.

3. Индукция по мощности множества. Иногда натуральный параметр характеризует сразу несколько объектов, рассматриваемых в задаче, и тогда при доказательстве по индукции необходимо варьировать все эти объекты. Чтобы найти их список, создан нормализатор "индуктпарам" (см. ниже). Ему передается терм "индуктпарам($\{x\}$)", где x - первоначально усмотренный объект с натуральным параметром n. Нормализатор преобразует его в терм "индуктпарам($\{x,y,\ldots\}$, добавляя прочие варьируемые объекты $y,\ldots,$. Прием докательства

индукцией по мощности конечного множества иллюстрирует схему действий в таких случаях:

$\forall_m(\text{индуктпарам}\{A\} = \text{индуктпарам}\{A;C\}\ \&\ m - \text{натуральное}\ \&\ \forall_{Anx}(A - \text{set}\ \&\ \text{card}A = m\ \&\ P(A,x) \to Q(A,m,x))\ \&\ \forall_{nBy}(n - \text{натуральное}\ \&\ 0 \leq n-m\ \&\ \forall_{Ax}(A-\text{set}\ \&\ \text{card}A = n\ \&\ P(A,x) \to Q(A,n,x))\ \&\ B-\text{set}\ \&\ P(B,y)\ \&\ \text{card}B = n+1 \to Q(B,n+1,y)) \to \forall_{Anx}(A - \text{set}\ \&\ n - \text{натуральное}\ \&\ 0 \leq n-m\ \&\ \text{card}A = n\ \&\ P(A,x) \to Q(A,n,x)))$

Заголовок приема - "свертка". Условие задачи на доказательство идентифицируется с утверждением $Q(A,n,x)$, некоторые из посылок - с утверждениями "$A - \text{set}, n - \text{натуральное}, 0 \leq n-m, \text{card}A = n, P(A,x)$". Фактически сначала идентифицируются первые четыре утверждения. Затем обрабатывается первый антецедент, выделенный указателем "идентификатор". Он пополняет исходный одноэлементный список варьируемых объектов $\{A\}$ некоторым списком объектов C. Далее, в соответствии с указателем "контекст(равно(наборпеременных(x23)параметры(x28)))", идентифицируется список x всех параметров термов списка C. После этого происходит идентификация утверждения $P(A,x)$ как конъюнкции всех оставшихся посылок задачи на доказательство, содержащих хотя бы одну переменную списка A,x. В остальном прием аналогичен двум предыдущим. Второй антецедент обрабатывается проверочным оператором, третий и четвертый (база и шаг индукции) - обрабатываются вспомогательными задачами на доказательство. Уровень срабатывания равен 7.

4. Нормализатор "индуктпарам".

   Пока в нормализатор входит всего три приема. Все они связаны с добавлением к списку варьируемых объектов функции, имеющей варьируемую область определения.

   $\forall_{ABCf}(\text{Отображение}(f,A,C) \to \text{индуктпарам}\{A;B\} \leftrightarrow \text{индуктпарам}\{A,f,;B\})$

   Если имеется посылка "Отображение(f,A,C)", причем переменная A входит в список объектов, зависящих от варьируемого при доказательстве по индукции параметра, то в этот же список включается и переменная f. Следующие два приема имеют аналогичный характер. Их антецеденты берутся в посылках.

   $\forall_{ABf}(f-\text{функция}\ \&\ \text{Dom}(f) = A \to \text{индуктпарам}\{A;B\} \leftrightarrow \text{индуктпарам}\{A,f;B\})$

   $\forall_{ABf}(\text{перестановка}(f,A) \to \text{индуктпарам}\{A;B\} \leftrightarrow \text{индуктпарам}\{A,f;B\})$

Выдача ответа задачи на описание, имеющей натуральную неизвестную

Используются следующие шаблоны ответов:

"явное(x14 набор(натуральное(x14))набор(не(равно(x14 x1)))пустоеслово)". Содержащие неизвестную n условия суть "натуральное(n)", а также некоторое (возможно, пустое) множество утверждений вида $\neg(n = a)$, где a не содержит n.

"явное(x14 набор(натуральное(x14)меньшеилиравно(x14 x2))набор(не(равно(x14 x1)))пустоеслово)". Задача имеет условия "натуральное(n)", "$n \leq a$" и произвольное число условий вида $\neg(n = b)$.

Кванторная импликация с натуральным параметром

Во всех приемах этого раздела используется указатель "внешнийквантор", блокирующий попытки идентифицировать кванторную импликацию с переходом к отрицанию в кванторе существования.

1. Усмотрение шага индукции.

   $\forall_f(f(1) \to \forall_n(n - \text{натуральное} \ \& \ f(n) \to f(n+1)) \leftrightarrow \forall_n(n - \text{натуральное} \to f(n)))$

   Переменная f функциональная. Прием усматривает кванторную импликацию, идентифицируемую как шаг доказательства по индукции утверждения $f(n)$. Антецедент, обрабатываемый вспомогательной задачей на доказательство, предпринимает попытку установить базис индукции. Затем кванторная импликация заменяется на утверждение о том, что $f(n)$ выполняется для всех натуральных n. Уровень срабатывания равен 1.

2. Контрапозиция с переходом к конечному отрезку натуральных чисел.

   $\forall_{Pn}(n - \text{натуральное} \to \forall_k(P(k) \ \& \ k - \text{натуральное} \to \neg(k \leq n)) \leftrightarrow \forall_k(k \in \{1, \ldots, n\} \to \neg(P(k))))$

   Переменная P функциональная. Утверждение $P(k)$ идентифицируется с конъюнкцией всех антецедентов заменяемой кванторной импликации, отличных от антецедента "k – натуральное". Антецедент теоремы приема обрабатывается проверочным оператором. Уровень срабатывания равен 0.

3. Усмотрение ложного консеквента.

   $\forall_{aP}(\neg(a - \text{натуральное}) \to \forall_n(n - \text{натуральное} \ \& \ P(n) \to n = a) \leftrightarrow \forall_n(n - \text{натуральное} \to \neg(P(n))))$

   Переменная P функциональная. Антецедент теоремы приема обрабатывается вспомогательной задачей на доказательство. Введен сильный ограничитель трудоемкости. Уровень срабатывания равен 1.

4. Отбрасывание несущественного антецедента, запрещающего рассмотрение единичного значения.

   $\forall_A(A(1) \to \forall_n(n - \text{натуральное} \ \& \ 0 \leq n - 2 \to A(n)) \leftrightarrow \forall_n(n - \text{натуральное} \to A(n)))$

   Антецедент обрабатывается вспомогательной задачей на доказательство. В качестве дополнительной посылки ей придается преобразуемая кванторная импликация. Введен сравнительно слабый ограничитель трудоемкости. Уровень срабатывания равен 2.

5. Устранение сдвига натурального параметра.

   $\forall_A(A(1) \to \forall_m(m - \text{натуральное} \to A(m+1)) \leftrightarrow \forall_m(m - \text{натуральное} \to A(m)))$

   Указатель "новаргумент(x26 x13 фикс)" проверяет, что консеквент преобразуемой кванторной импликации содержит переменную m только в виде "$m+1$". Антецедент теоремы приема проверяет, что этот консеквент верен, если вместо $m+1$ подстаить единицу. Для этого используется вспомогательная задача на доказательство. Введен средний ограничитель трудоемкости. Уровень срабатывания равен 1.

Решение рекуррентных соотношений

1. Усмотрение экспоненциальной зависимости.

   $\forall_{afk}(k$ – целое $\to \forall_n(n$ – натуральное & $k \leq n \to f(n) = af(n-1)) \leftrightarrow \forall_n(n$ – натуральное & $k-1 \leq n \to f(n) = f(k-1)a^{n-k+1}))$

   Антецедент обрабатывается проверочным оператором. Уровень срабатывания равен 2.

2. Линейное рекуррентное соотношение первого порядка с переменными параметрами.

   $\forall_{fghk}(k$ – целое $\to \forall_n(n$ – натуральное & $k \leq n \to f(n) = g(n)f(n-1) + h(n)) \leftrightarrow \forall_n(n$ – натуральное & $k-1 \leq n \to f(n) = \sum_{i=k}^{n}\Big(h(i)\prod_{j=i+1}^{n} g(j)\Big) + \prod_{j=k}^{n} g(j)f(k-1)))$

   $\forall_{fghkm}(k$ – целое & m – целое $\to \forall_n(n \in \{k,\dots,m\} \to f(n) = (g(n)f(n-1) + h(n))/p(n)) \leftrightarrow \forall_n(n \in \{k-1,\dots,m\} \to f(n) =$
   $\sum_{i=k}^{n}((h(i)/p(i))\prod_{j=i+1}^{n} g(j)/p(j)) + \prod_{j=k}^{n}(g(j)/p(j))f(k-1)))$

   Антецеденты обрабатываются проверочными операторами. Коэффициенты $h(n)$, $g(n)$ не должны иметь вхождения символа "определитель". Уровень срабатывания равен 2.

3. Рекуррентное линейное соотношение второго порядка, дающее линейное решение.

   $\forall_{akmp}(k$ – целое & $m = a(k-1) - a(k-2)$ & $p = a(k-2) - (k-2)m \to \forall_n(n$ – натуральное & $k \leq n \to a(n) = 2a(n-1) - a(n-2)) \leftrightarrow \forall_n(n$ – натуральное & $k-2 \leq n \to a(n) = p + mn))$

   Второй и третий антецеденты выделены указателем "идентификатор". Их правые части обрабатываются нормализаторами общей стандартизации. Первый антецедент обрабатывается проверочным оператором. Уровень срабатывания равен 1.

4. Общий случай линейного однородного уравнения с постоянными коэффициентами. Прием реализует стандартную схему действий: строится характеристический многочлен, находится разложение его на множители, затем с помощью вспомогательного синтезатора "Базисрешений" (см. ниже) по найденным множителям определяются выражения для базисных функций пространства решений однородного уравнения, и наконец выписывается соотношение для общего решения.

   $\forall_{adkmpry}\neg(a(m+1) = 0)$ & $p = \sum_{j=1}^{m+1}(a(j)n^{j-1})$ & Базисрешений(p,d) & $d = \{;r\}$ & $l(r) = k \to \forall_n\Big(n$ – целое & $s \leq n \to \sum_{j=1}^{m+1}(a(j)y(n+j-1)) = 0\Big) \leftrightarrow$ $\exists_c(\forall_n\Big(n$ – целое & $s \leq n \to y(n) = \sum_{i=1}^{k}(c(i)r(i))\Big)$ & $\forall_i(i \in \{1,\dots,k\} \to c(i)$ – число))

   Кванторная импликация, определяющая рекуррентную зависимость, является условием задачи на описание. Переменная y - неизвестная этой задачи, m - натуральная константа. Указатель "развертка" определяет идентификацию конечной суммы $\sum_{j=1}^{m+1}(a(j)y(n+j-1))$ с обычной суммой. При этом указатель "подтерм(фикс(0 1 6 1 1 3 2 2))" уточняет, что выражения $n+j-1$ сравниваются с термами, независимо построенными по известным на момент идентификации n, j. Эти термы обрабатываются нормализатором "нормплюс". Чтобы заблаговременно идентифицировать m, используется указатель "контекст(позиция(х5 теквхожд)вид(х5 значение(х24 плюс(х14 х13))))". Таким образом, до идентификации суммы в ней будет усматриваться подвыражение $y(n+m)$ для "максимального" m. Введен также указатель "коэфф(х1)", определяющий идентификацию a с набором коэффициентов при $y(n), \ldots, y(n+m)$. Если в сумме отсутствовал член с $y(n+i)$, то $a(i)$ будет полагаться равным нулю. Первый антецедент, обрабатываемый проверочным оператором, устанавливает отличие коэффициента при $y(n+m)$ от нуля. Второй антецедент, выделенный указателем "идентификатор", строит характеристический многочлен. Конечная сумма в правой его части выделена указателем "развертка", т.е. выписывается как обычная сумма. Многочлен обрабатывается нормализаторами "видУмножение", "нормвеществ", "Корни". Первый из них находит разложение многочлена на комплексные множители, второй - устраняет комплекснозначные операции над вещественными числами, заменяя их на вещественнозначные аналоги, третий - преобразует разложение к такому виду, чтобы переменая n имела в каждом множителе коэффициент единица. При этом коэффициенты пропорциональности отбрасываются. Подробнее данные нормализаторы будут рассмотрены в разделе, посвященном комплексным числам. Третий антецедент обрабатывается синтезатором "Базисрешений", описываемым ниже. Четвертый антецедент выделен указателем "идентификатор". Он идентифицирует результат d работы синтезатора как конечное множество, перечисляемое набором r. Пятый антецедент тоже выделен указателем "идентификатор" - он находит длину k набора r с помощью нормализатора "нормдлинанабора". Перед построением параметрического описания для общего решения создается набор c новых переменных, длина которого равна k. Это обеспечивается указателем "переменные(х3 х11)". Конечная сумма для линейной комбинации базисных решений и кванторная импликация $\forall_i(i \in \{1, \ldots, k\} \to\ c(i) -$ число$)$ выделены указателем "развертка". Они выписываются как обычная сумма и конъюнкция утверждений "$c(i) -$ число". Уровень срабатывания равен 2. Аналогичный прием создан для преобразования рекуррентности вида

$$\forall_n \left(n - \text{натуральное} \ \& \ s \leq n \to \textstyle\sum_{j=1}^{m+1}(a(j)y(n+j-1)) = 0\right)$$

5. Решение линейного неоднородного рекуррентного уравнения с постоянными коэффициентами при помощи синтезатора "линчастнреш". Для нахождения частного решения используется синтезатор "линчастнреш" (см. ниже), в остальном прием аналогичен случаю однородного уравнения:

   $\forall_{adkmpruvy} \neg(a(m+1) = 0) \ \& \ p = \sum_{j=1}^{m+1}(a(j)n^{j-1}) \ \& \ \text{Базисрешений}(p, d) \ \& \ d = \{; r\} \ \& \ l(r) = k \ \& \ \text{линчастнреш}(a, n, u(n), v) \to \forall_n(n - \text{натуральное} \ \& \ s \leq n \to \sum_{j=1}^{m+1}(a(j)y(n+j-1)) = u(n)) \leftrightarrow \exists_c(\forall_n(n - \text{натуральное} \ \& \ s \leq n \to y(n) = v + \sum_{i=1}^{k}(c(i)r(i))) \ \& \ \forall_i(i \in \{1, \ldots, k\} \to\ c(i) - \text{число}))$

   Если $u(n)$ идентифицируется с нулем, то прием блокируется. Уровень срабатывания равен 2.

6. Группировка ненулевых членов рекуррентного уравнения в одной части.

   $\forall_{fg}(\forall_n(n - \text{натуральное} \to f(n) = g(n)) \leftrightarrow \forall_n(n - \text{натуральное} \to f(n) - g(n) = 0))$

Группировка предпринимается для того, чтобы могли сработать описанные выше приемы, у которых правая часть уравнения была нулевая. Кванторая импликация входит в условие задачи на описание. Выражения $f(n), g(n)$ оба содержат как неизвестную, так и параметр n. Ни одно из них не имеет вида $x(n)$, где x - неизвестная. Переменные f, g - функциональные. Уровень срабатывания равен 2.

7. Решение рекуррентности в посылках.

   $\forall amyF((\forall_n(n - \text{натуральное} \to F(y,n))\ \&\ \text{последовательность}(y,R)\ \&\ \forall_i(i \in \{1,\ldots,m\} \to y(i) = a(i))\ \&\ y - \text{функция}) = B(y) \to \forall_n(n - \text{натуральное} \to F(y,n))\ \&\ \text{последовательность}(y,R)\ \&\ \forall_i(i \in \{1,\ldots,m\} \to y(i) = a(i))\ \&\ y - \text{функция} \leftrightarrow B(y))$

   Прием имеет заголовок "заменатермов(второйтерм)". Утверждения "последовательность(y, R)", "$\forall_n(n -$ натуральное $\to F(y,n))$", "$\forall_i(i \in \{1,\ldots,m\} \to y(i) = a(i))$", "$y -$ функция" берутся в посылках. Они образуют рекуррентное задание обозначенной переменной y функции. Переменные a, F - функциональные. Кванторная импликация $\forall_i(i \in \{1,\ldots,m\} \to y(i) = a(i))$ выделена указателем "развертка", т.е. идентифицируется с группой посылок вида $y(i) = a(i)$. Антецедент выделен указателем "идентификатор". Его левая часть обрабатывается вспомогательной задачей на описание, имеющей единственную неизвестную y. При решении этой задачи используется описанный выше аппарат решения рекуррентных соотношений, и результат $B(y)$ замещает исходную группу посылок. Уровень срабатывания равен 4.

8. Синтезатор "Базисрешений". Обращение к синтезатору имеет вид "Базисрешений(a, b)". Здесь a - характеристический многочлен, разложенный на линейные множители, причем коэффициент при переменной в каждом множителе равен 1. В качестве b выдается терм для конечного множества выражений, определяющих базисные решения. Чтобы определить переменную x, относительно которой рассматривается многочлен a, при обращении оператору передается комментарий (переменная x). Синтезатор имеет следующие приемы:

   (a) Константный множитель.

   $\forall_a(\text{Базисрешений}(a, \emptyset))$

   Выражение a - константное.

   (b) Случай вещественного корня.

   $\forall_{abcnx}(a - \text{число}\ \&\ \text{Базисрешений}(b,c) \to \text{Базисрешений}((x+a)^n b, \text{set}_y(\exists_i(i \in \{0,\ldots,n-1\}\ \&\ y = x^i(-a)^x)) \cup c))$

   Созданы два приема, один для комплекснозначного произведения, другой - для вещественнозначного (если все корни характеристического многочлены оказались вещественными). Теоремы этих приемов прорисовываются формульным редактором одинаково, но в первом случае используется символ "Умножение", а во втором - "умножение". Показатель степени n - натуральная константа. Первый антецедент обрабатывается проверочным оператором, второй - реализует рекурсивное обращение к синтезатору для обработки остатка произведения. Указатель "или" определяет развертку квантора существования в дизъюнкцию равенств $y = (-a)^x, \ldots, y =$

$x^{n-1}(-a)^x$. Нормализатор "нормкласс" далее преобразует полученное выражение $\mathrm{set}_y(y = (-a)^x \vee \ldots \vee y = x^{n-1}(-a)^x)$ в конечный перечень $\{(-a)^x, \ldots, x^{n-1}(-a)^x\}$.

(c) Случай комплексного корня.

$\forall_{abcdnqx}(a - \text{число}\ \&\ b - \text{число}\ \&\ \text{Базисрешений}(c, d)\ \&\ q = \sqrt{a^2 + b^2}\ \&\ \neg(q = 0) \to \text{Базисрешений}((x + a + bi)^n c, \mathrm{set}_y(\exists_k(k \in \{0, \ldots, n-1\}\ \&\ y = x^k q^x \cos(x \arccos(a/q)))) \cup d))$

$\forall_{abcdnqx}(a - \text{число}\ \&\ b - \text{число}\ \&\ \text{Базисрешений}(c, d)\ \&\ q = \sqrt{a^2 + b^2}\ \&\ \neg(q = 0) \to \text{Базисрешений}((x + a - bi)^n c, \mathrm{set}_y(\exists_k(k \in \{0, \ldots, n-1\}\ \&\ y = x^k q^x \sin(x \arccos(a/q)))) \cup d))$

Аналогично случаю вещественного корня.

(d) Условное выражение.

$\forall_{abcdP}(\text{Базисрешений}(a, c)\ \&\ \text{Базисрешений}(b, d) \to \text{Базисрешений}((a\ \text{при}\ P, \text{иначе}\ b), (c\ \text{при}\ P, \text{иначе}\ d)))$

Антецеденты реализуют рекурсивные обращения.

9. Синтезатор "Линчастнреш". Обращение к синтезатору имеет вид "линчастнреш(a, x, b, c)". Здесь a - терм, определяющий набор коэффициентов рекуррентного соотношения, x - переменная для номера элемента определяемой последовательности, b - ненулевая правая часть рекуррентного соотношения. В качестве c выдается какое-либо частное решение.

(a) Независимое рассмотрение слагаемых правой части рекуррентного соотношения.

$\forall_{abcdex}(\text{линчастнреш}(a, x, b, d)\ \&\ \text{линчастнреш}(a, x, c, e) \to \text{линчастнреш}(a, x, b + c, d + e))$

Антецеденты реализуют рекурсивные обращения к синтезатору.

(b) Степенная функция.

$\forall_{acdefmnpqx}(m = l(a)\ \&\ f = \sum_{j=1}^{m+1}(a(j)x^{j-1})\ \&\ f = (x-1)^p e\ \&\ (\forall_i(i \in \{1, \ldots, n+1\} \to q(i) - \text{число})\ \&\ \sum_{j=1}^{m+1}(a(j)\sum_{i=0}^{n}(q(i+1)(x+j-1)^{i+p})) = cx^n) = \forall_i(i \in \{1, \ldots, n+1\} \to q(i) = d(i)) \to \text{линчастнреш}(a, x, cx^n, \sum_{i=0}^{n}(d(i+1)x^{i+p})))$

Указатель "подстановка(фикс(0 3 2)x14)" разрешает показателю степени n обращаться в 0. Антецеденты выделены указателем "идентификатор". Первый, используя нормализатор "нормдлинанабора", определяет длину m набора коэффициентов a. Второй строит характеристический многочлен, предпринимает попытку разложения его на вещественные множители и обращается к нормализатору "Корни", чтобы сделать все коэффициенты при x в линейных множителях равными 1. Третий антецедент выделяет в найденном произведении f множитель вида $(x-1)^p$, причем указатель "подстановка(фикс(3 2 1)x15 0)" разрешает нулевой показатель степени p. Четвертый антецедент решает систему уравнений для нахождения коэффициентов $q(i)$ частного решения. Предварительно, согласно указателю "переменные(x16 плюс(x14 1))", вводится набор новых переменных $q(1), \ldots, q(n+1)$. Указатели "развертка" определяют преобразование кванторной импликации и конечных сумм при построении системы уравнений в конъюнкцию и обычные суммы. Для решения системы используется задача на описание с неизвестными $q(1), \dots, q(n{+}1)$, имеющая дополнительную цель (независит x), запрещающую переменной x входить в ответ.

(c) Экспонента.

$\forall_{acdefmnpqx}(m = l(a)\ \&\ f = \sum_{j=1}^{m+1}(a(j)x^{j-1})\ \&\ f = (x-k)^p e\ \&\ (\forall_i(i \in \{1,\dots,n{+}1\} \rightarrow q(i)-\text{число})\ \&\ \sum_{j=1}^{m+1}(a(j)\sum_{i=0}^{n}(q(i{+}1)(x{+}j{-}1)^{i+p}k^{x+j-1})) = cx^n k^x) = \forall_i(i \in \{1,\dots,n{+}1\} \rightarrow q(i) = d(i))\ \&\ r = cx^n \rightarrow \text{линчастнреш}(a, x, rk^x, \sum_{i=0}^{n}(d(i+1)x^{i+p}k^x)))$

Аналогично предыдущему. Добавляется пятый антецедент, который тоже выделен указателем "идентификатор". Он проверяет, что множитель r представляет собой произведение не содержащего x выражения c на некоторую (возможно, нулевую) степень x.

Усмотрение истинности либо ложности условия "натуральное(n)"

Утверждение "натуральное(a)" может быть заменено на логическую константу "истина" либо "ложь" следующими двумя приемами, имеющими заголовок "второйтерм":

$\forall_a(a - \text{натуральное} \rightarrow a - \text{натуральное})$

$\forall_a(\neg(a - \text{натуральное}) \rightarrow \neg(a - \text{натуральное}))$

Антецеденты приемов обрабатываются, соответственно, проверочными операторами "усмнатуральное" и "усмненатуральное"(см. ниже). Первый прием блокируется, если утверждение "натуральное(a)" является условием задачи на описание, причем a - неизвестная, для которой в условиях нет равенства, выражающего ее через другие неизвестные. Указатель "новый" отменяет блокировку повторной попытки применения приема при возвращении утверждения в зону внимания. Первый прием срабатывает на уровне 2, для второго этот уровень в случае условия задачи на описание повышается до 4.

Проверочный оператор "усмнатуральное"

Кроме приема, усматривающего наличие в посылках оператора проверяемого утверждения "натуральное(a)", созданы следующие приемы:

1. Сумма.

   $\forall_{ab}(a - \text{натуральное}\ \&\ b - \text{натуральное} \rightarrow (a+b) - \text{натуральное})$

   Антецеденты реализуют рекурсивное обращение к оператору. Указатель "дистрибразвертка(фикс(0 1))" определяет компиляцию приема с обработкой суммы произвольного числа слагаемых путем независимого обращения к усмотрению натуральности каждого из них.

   $\forall_{abn}(n = a + b\ \&\ a - \text{целое}\ \&\ 0 \leq a\ \&\ b - \text{натуральное} \rightarrow n - \text{натуральное})$

   Первый антецедент берется в посылках, остальные - обрабатываются проверочными операторами.

2. Произведение.

   $\forall_{ab}(a$ – натуральное & b – натуральное $\to ab$ – натуральное)

   Как и в случае суммы, используется указатель "дистрибразвертка(фикс(0 1))".

3. Степень.

   $\forall_{ab}(a$ – натуральное & b – целое & $0 \leq b \to a^b$ – натуральное)

   Антецеденты обрабатываются проверочными операторами.

4. Целое число, большее нуля. Предшествующие приемы имели уровень срабатывания 1. Уровень срабатывания данного приема уже равен 2:

   $\forall_n(n$ – целое & $0 \leq n - 1 \to n$ – натуральное)

   Антецеденты обрабатываются проверочными операторами.

5. Факториал.

   $\forall_n(n$ – целое & $0 \leq n \to n!$ – натуральное)

   Уровень срабатывания равен 1.

6. Принадлежность конечному отрезку натуральных чисел.

   $\forall_{mnk}(k \in \{m, \ldots, n\}$ & m – натуральное $\to k$ – натуральное)

   Первый антецедент берется в посылках, второй - обрабатывается проверочным оператором. Уровень срабатывания равен 2.

7. Элемент последовательности натуральных чисел.

   $\forall_{ai}(i$ – натуральное & последовательность$(a, \mathbf{N}) \to a(i)$ – натуральное)

   Антецеденты обрабатываются проверочными операторами. Уровень срабатывания равен 1.

8. Использование кванторной импликации в посылках.

   $\forall_{afP}(P(a)$ & $\forall_x(P(x) \to f(x)$ – натуральное$) \to f(a)$ – натуральное)

   Второй антецедент берется в посылках, первый - обрабатывается вспомогательной задачей на доказательство. Переменная P функциональная. Уровень срабатывания равен 2.

9. Простое число.

   $\forall_n($простое$(n) \to n$ – натуральное)

   Антецедент берется в посылках. Уровень срабатывания равен 3.

10. Функция, принимающая натуральные значения.

    $\forall_{fi}(\mathrm{Val}(f) \subseteq \mathbf{N} \to f(i)$ – натуральное)

    Антецедент обрабатывается проверочным оператором. Переменная f - обычная. Уровень срабатывания равен 1.

11. Мощность непустого конечного множества.

    $\forall_A(\mathrm{card}A = n$ & $\neg(A = \emptyset)$ & n – число $\to n$ – натуральное)

    Первый антецедент берется в посылках, второй - обрабатывается проверочным оператором. Уровень срабатывания равен 2.

12. Конечный порядок элемента группы.

    $\forall_{Aaf}$(группа(A) & $f =$ операция(A) & $a \in$ носитель(A) & порядокэлемента$(a, A)-$ число $\rightarrow$ порядокэлемента(a, A) – натуральное)

    Первый антецедент берется в посылках, второй - выделен указателем "идентификатор", два последних - обрабатываются проверочными операторами. Уровень срабатывания равен 1.

13. Элемент набора натуральных чисел.

    $\forall_{ain}$(кортеж$(a, n, \mathbf{N})$ & $i \in \{1, \ldots, n\} \rightarrow a(i)$ – натуральное)

    Первый антецедент берется в посылках, второй обрабатывается проверочным оператором. Уровень срабатывания равен 1.

Проверочный оператор "усмненатуральное"

Кроме приема непосредственного извлечения проверяемого утверждения "$\neg$(натуральное(a))"из посылок, оператор имеет всего три приема:

1. Нецелое число.

   $\forall_n(\neg(n$ – целое$) \rightarrow \neg(n$ – натуральное$))$

   Антецедент обрабатывается проверочным оператором "усмнецелое".

2. Нерациональное число.

   $\forall_a(\neg(a - \mathrm{rational}) \rightarrow \neg(a$ – натуральное$))$

   Антецедент обрабатывается проверочным оператором "усмнерациональное".

3. Неположительное число.

   $\forall_n(n \leq 0 \rightarrow \neg(n$ – натуральное$))$

   Антецедент обрабатывается проверочным оператором. Введен сильный ограничитель трудоемкости.

13.3 Приемы символа "рациональное"

Утверждение "рациональное(a)", прорисовываемое формульным редактором в виде "$a - \mathrm{rational}$", означает, что a есть рациональное число. Символ "рациональное" характеризуется справочниками "арность", "предикатныйсимвол", "род", "родобъекта". Он задает один из основных типов объектов, причем надтипами являются типы "число", "комплексное".

Общая стандартизация утверждений

1. Отбрасывание минуса.

   $\forall_a((-a) - \mathrm{rational} \leftrightarrow a - \mathrm{rational})$

   Уровень срабатывания равен 2.

2. Отбрасывание рационального слагаемого.

   $\forall_{ab}(a - \text{rational} \to (a+b) - \text{rational} \leftrightarrow b - \text{rational})$

   Антецедент обрабатывается проверочным оператором. Уровень срабатывания равен 2.

3. Дробное выражение.

   $\forall_{pq}(q - \text{rational}\ \&\ \neg(q=0) \to (p/q) - \text{rational} \leftrightarrow p - \text{rational})$

   Антецеденты обрабатываются проверочными операторами. Уровень срабатывания равен 0.

   $\forall_{a}((1/a) - \text{rational} \leftrightarrow a - \text{rational})$

   Уровень срабатывания равен 2.

4. Отбрасывание рационального сомножителя.

   $\forall_{ab}(a - \text{rational}\ \&\ \neg(a=0) \to (ab) - \text{rational} \leftrightarrow b - \text{rational})$

   Антецеденты обрабатываются проверочными операторами. Уровень срабатывания равен 1.

5. Раскрывание скобок.

   $\forall_{ab}(b = a \to a - \text{rational} \leftrightarrow b - \text{rational})$

   Выражение a имеет сомножитель вида $(A+B)^n$, где n - натуральная константа, равная 1, 2 либо 3. Антецедент выделен указателем "идентификатор". Его правая часть обрабатывается нормализатором раскрывания скобок "стандплюс". Уровень срабатывания равен 2.

6. Попытка разложения на множители для усмотрения произведения нерационального числа на рациональное.

   $\forall_{abc}(a = bc\ \&\ \neg(c - \text{rational}) \to a - \text{rational} \leftrightarrow b = 0)$

   Выражение a должно иметь заголовок "плюс". Первый антецедент выделен указателем "идентификатор". Его левая часть обрабатывается нормализатором "видумножение". Указатель "перечень(x2 легковидеть(рациональное(x2)))" определяет идентификацию переменной b как произведение всех сомножителей результата разложения, для которых очевидна рациональность. Проверяется, что это произведение не равно 1. Второй антецедент обрабатывается проверочным оператором. Уровень срабатывания равен 2.

7. Иррациональность числа "пи".

   $\neg(\pi - \text{rational})$

   Прием имеет заголовок "второйтерм". Уровень срабатывания равен 0.

Исключение квантора

$\exists_{a}(a - \text{rational})$

$\forall_{a}(\exists_{x}(x - \text{rational}\ \&\ \neg(x = a)\ \&\ \text{числитель}(x) - \text{even}))$

Приемы имеют заголовок "второйтерм". Уровень срабатывания равен 0.

$\forall_x(\exists_{pq}(x = pq \;\&\; p-\text{rational} \;\&\; q-\text{rational} \;\&\; \text{числитель}(p)-\text{even} \;\&\; \neg(\text{знаменатель}(q)-\text{even})) \leftrightarrow x-\text{rational} \;\&\; \text{числитель}(x)-\text{even})$

Уровень срабатывания равен 1.

$\forall_{an}(\exists_x(x-\text{rational} \;\&\; \forall_i(i \in \{1,\dots,n\} \rightarrow \neg(x = a(i)))))$

Указатель "развертка(фикс(0 2 2))" определяет идентификацию кванторной импликации с конъюнкцией утверждений $\neg(x = a(1)), \dots, \neg(x = a(n))$. Переменная a - функциональная. Уровень срабатывания равен 1.

$\forall_{abP}(a(y)-\text{rational} \;\&\; b(y)-\text{rational} \;\&\; \neg(a(y) = 0) \rightarrow \exists_{xy}(x-\text{rational} \;\&\; a(y)x = b(y) \;\&\; P(x, y)) \leftrightarrow \exists_y(P(b(y)/a(y), y)))$

Переменные a, b, P функциональные. Антецеденты обрабатываются проверочными операторами. При каждом обращении добавляется посылка $P(x, y)$. Указатель "кортежпеременных(x24)" разрешает идентификацию y с набором переменных, имеющим произвольную длину. Фильтр "не(заголовок(x24 пустоеслово))" проверяет, что этот набор непуст. Уровень срабатывания равен 2.

Усмотрение рационального значения неизвестной, отличного от заданных чисел

$\forall_{an}(\exists_x(x-\text{rational} \;\&\; \forall_i(i \in \{1,\dots,n\} \rightarrow \neg(x = a(i)))))$

$\forall_{abn}(b-\text{число} \rightarrow \exists_x(x-\text{rational} \;\&\; \forall_i(i \in \{1,\dots,n\} \rightarrow \neg(x = a(i))) \;\&\; b < x))$

Приемы имеют заголовок "связка". Квантор общности выделен указателем "развертка". Утверждения под квантором существования идентифицируются со всеми условиями задачи на описание, содержащими несущественную неизвестную x. Во втором приеме допускаются случай нестрогого неравенства и перестановка частей неравенства. Уровень срабатывания равен 3.

Усмотрение множества рациональных чисел

$\text{set}_z(z-\text{rational}) = \mathbf{Q}$

Уровень срабатывания равен 1.

Выдача ответа задачи на описание, имеющей рациональную неизвестную

Используется следующий шаблон ответа:

"явное(x14 набор(рациональное(x14))набор(не(равно(x14 x1)))пустоеслово)".

Усмотрение истинности либо ложности условия "рациональное(p)"

Утверждение "рациональное(a)" может быть заменено на логическую константу "истина" либо "ложь" следующими двумя приемами, имеющими заголовок "второйтерм":

$\forall_a(a-\text{рациональное} \rightarrow a-\text{рациональное})$

$\forall_a(\neg(a-\text{рациональное}) \rightarrow \neg(a-\text{рациональное}))$

Антецеденты приемов обрабатываются, соответственно, проверочными операторами "усмрациональное" и "усмнерациональное"(см. ниже). Приемы срабатывают только в условиях задач. Уровень срабатывания первого приема равен 2. Для содержащего неизвестные условия задачи на описание уровень срабатывания второго приема повышается до 4.

Проверочный оператор "усмрациональное"

1. Натуральное число.

   $\forall_n(n - \text{натуральное} \to n - \text{rational})$

   Антецедент берется в посылках. Выражение n неконстантное.

2. Целое число.

   $\forall_n(n - \text{целое} \to n - \text{rational})$

   Аналогично предыдущему.

3. Отбрасывание минуса.

   $\forall_a(a - \text{rational} \to (-a) - \text{rational})$

   Антецедент обрабатывается проверочным оператором. Указатель "спуск" блокирует альтернативные попытки.

4. Сумма рациональных чисел.

   $\forall_{ab}(a - \text{rational} \ \& \ b - \text{rational} \to (a+b) - \text{rational})$

   Антецеденты обрабатываются проверочными операторами. Указатель "дистрибразвертка(фикс(0 1))" определяет одновременную обработку всех слагаемых.

5. Произведение рациональных чисел.

   $\forall_{ab}(a - \text{rational} \ \& \ b - \text{rational} \to ab - \text{rational})$

   Аналогично предыдущему.

6. Дробь с рациональными числителем и знаменателем.

   $\forall_{ab}(\neg(b = 0) \ \& \ a - \text{rational} \ \& \ b - \text{rational} \to (a/b) - \text{rational})$

   Антецеденты обрабатываются проверочными операторами.

7. Целочисленная степень рационального числа.

   $\forall_{an}(n - \text{целое} \ \& \ a - \text{rational} \to a^n - \text{rational})$

   Аналогично предыдущему.

8. Использование кванторной посылки.

   $\forall_{axP}(x \in a \ \& \ a \subseteq \text{set}_y(y - \text{rational} \ \& \ P(y)) \to x - \text{rational})$

   Переменная P - функциональная. Оба антецедента берутся в посылках.

9. Проверка принадлежности множеству рациональных чисел.

   $\forall_a(a \in \mathbf{Q} \to a - \text{rational})$

   Антецедент обрабатывается проверочным оператором.

Проверочный оператор "усмнерациональное"

1. Отбрасывание минуса.

   $\forall_a(\neg(a - \text{rational}) \to \neg((-a) - \text{rational}))$

   Антецедент обрабатывается проверочным оператором.

2. Отбрасывание рационального слагаемого.

   $\forall_{ab}(b - \text{rational} \ \&\ \neg(a - \text{rational}) \to \neg((a+b) - \text{rational}))$

   Антецеденты обрабатываются проверочными операторами.

3. Отбрасывание ненулевого рационального сомножителя.

   $\forall_{ab}(\neg(a = 0) \ \&\ a - \text{rational} \ \&\ \neg(b - \text{rational}) \to \neg(ab - \text{rational}))$

   Аналогично предыдущему.

13.4 Приемы символа "числитель"

Выражение "числитель(a)" обозначает числитель несократимой дроби, представляющей рациональное число a. Числитель берется со знаком числа a. Формульный редактор прорисовывает это выражение в том же самом виде "числитель(a)". Символ "числитель" характеризуется справочниками "арность", "тип", "типданных".

Изменение знака

$\forall_a(a - \text{rational} \to \text{числитель}(-a) = -\text{числитель}(a))$

Уровень срабатывания равен 0.

Обратная дробь

$\forall_a(0 < a \to \text{числитель}(1/a) = \text{знаменатель}(a))$

Антецедент обрабатывается проверочным оператором. Уровень срабатывания приема равен 1.

Соотношение пропорциональности для числителя и знаменателя

$\forall_{abx}(a \cdot \text{числитель}(x) = b \cdot \text{знаменатель}(x) \leftrightarrow ax = b)$

Уровень срабатывания равен 0.

Нормализатор общей стандартизации "нормчислитель"

Нормализатор имеет реализованный на ЛОСе прием, находящий числители десятичных дробей. Кроме того, продублированы приемы для изменения знака и обратной дроби. Добавлен прием, определяющий числитель целого числа: $\forall_a(a - \text{целое} \to \text{числитель}(a) = a)$

13.5 Приемы символа "знаменатель"

Выражение "знаменатель(a)" обозначает знаменатель несократимой дроби, представляющей рациональное число a. Знаменатель всегда берется со знаком "плюс". Формульный редактор прорисовывает это выражение в том же самом виде "знаменатель(a)". Символ "знаменатель" характеризуется справочниками "арность", "тип", "типданных".

Знаменатель целого числа

$\forall_n(n - \text{целое} \to \text{знаменатель}(n) = 1)$

Антецедент обрабатывается проверочным оператором. Уровень срабатывания приема равен 0.

Отбрасывание минуса

$\forall_a(\text{знаменатель}(-a) = \text{знаменатель}(a))$

Уровень срабатывания равен 0.

Обратная дробь

$\forall_a(0 < a \to \text{знаменатель}(1/a) = \text{числитель}(a))$

Уровень срабатывания равен 1.

Нормализатор общей стандартизации "нормзнаменатель"

Нормализатор имеет реализованный на ЛОСе прием, находящий знаменатели десятичных дробей. Кроме того, продублированы приемы для знаменателя целого числа, изменения знака и обратной дроби.

13.6 Приемы символа "делит"

Утверждение "делит(a b)", прорисовываемое формульным редактором в виде "$a|b$, означает, что a, b суть целые числа, причем a делит b. Символ "делит" характеризуется справочниками "арность", "предикатныйсимвол", "одз", "типданных", "транзитивно".

Общая стандартизация утверждений

1. Отбрасывание слагаемого, делящегося на рассматриваемое выражение.

   $\forall_{abc}(a|b \to a|(b+c) \leftrightarrow a|c)$

   Антецедент обрабатывается проверочным оператором. Уровень срабатывания равен 1.

2. Отбрасывание минуса.

   $\forall_{mn}(n|m \leftrightarrow n|(-m))$

   $\forall_{mn}(m|n \leftrightarrow (-m)|n)$

Замены выполняются справа налево. Уровень срабатывания равен 2.

3. Учет взаимной простоты.

   $\forall_{mnk}($взаимнопросты$(m,n) \to n|(mk) \leftrightarrow n|k)$

   Антецедент обрабатывается проверочным оператором. Уровень срабатывания равен 3.

4. Сокращение на общий целочисленный ненулевой множитель.

   $\forall_{mnk}(m -$ целое $\&\ n -$ целое $\&\ k -$ целое $\&\ \neg(k=0) \to (mk)|(nk) \leftrightarrow m|n)$

   Антецеденты обрабатываются проверочными операторами. Допускается обращение m, n в единицу, но k не должно равняться единице. Уровень срабатывания равен 1.

5. Делимость на 2 произведения двух целых чисел, разность которых нечетна.

   $\forall_{mnkp}(\neg(m -$ even$)\ \&\ \neg((n-k) -$ even$)\ \&\ m -$ целое $\&\ n -$ целое $\&\ k -$ целое $\to (2m)|(knp) \leftrightarrow m|(knp))$

   Антецеденты обрабатываются проверочными операторами. Требуется, чтобы выражения k, n имели вид $x+a, x+b$, где a, b - константы (возможно, нулевые); x - неконстантное. Множители m, p могут обращаться в единицу. Уровень срабатывания равен 1.

6. Делимость модуля целого числа.

   $\forall_{mn}(m||n| \leftrightarrow m|n)$

   Уровень срабатывания равен 0.

7. Свертка двух условий делимости в одно.

   $\forall_{mnk}(m|k\ \&\ n|k \leftrightarrow$ нок$(m,n)|k)$

   Переменные m, k идентифицируются с натуральными константами. Уровень срабатывания равен 1.

   $\forall_{mnk}(m -$ целое $\&\ n -$ целое $\to m|k\ \&\ n|k \leftrightarrow$ нок$(m,n)|k)$

   Антецеденты берутся в контексте. Преобразуемая конъюнкция заключена внутри описателя, имеющего связанную переменную k. Уровень срабатывания равен 1.

8. Делители единицы.

   $\forall_{m}(m -$ натуральное $\to m|1 \leftrightarrow m=1)$

   Антецедент обрабатывается проверочным оператором. Уровень срабатывания равен 0.

   $\forall_{mn}((m+1) -$ натуральное $\&\ n -$ целое $\to \forall_x(x \in \{0,\dots,m\} \to n|x) \leftrightarrow m = 0\ \vee\ |n| = 1)$

   $\forall_{mn}((m+1) -$ натуральное $\&\ n -$ целое $\to \forall_x(x -$ целое $\&\ 0 \le x\ \&\ x \le m \to n|x) \leftrightarrow m = 0\ \vee\ |n| = 1)$

   Антецеденты обрабатываются проверочными операторами. Уровень срабатывания равен 1.

9. Делимость конечной суммы.

   Утверждение о делимости конечной суммы на натуральную константу m заменяется утверждением о делимости на m суммы произведений всевозможных вычетов по модулю m на числа слагаемых, имеющих данный вычет:

   $\forall_{Amt}(m|\sum_{i,A(i)} t(i) \leftrightarrow m|\sum_{j=1}^{m-1}(j\mathrm{card}(\mathrm{set}_i(A(i)\ \&\ t(i)(\mathrm{mod}\ m) = j))))$

   Преобразуемое утверждение входит в условие задачи. Оно расположено внутри выражения "мощность(...)". Переменная m идентифицируется с натуральной константой. Переменные t, A - функциональные, причем связывающая приставка i может быть произвольной длины. Уровень срабатывания равен 4.

10. Делимость на дробь

    $\forall_{mn}(m|n \to (n/m)|n)$

    Прием имеет заголовок "второйтерм". Антецедент обрабатывается проверочным оператором. Уровень срабатывания равен 1.

11. Решение вспомогательной задачи на вычисление для явного определения натуральной константы.

    $\forall_{mnk}(k = m \to n|m \leftrightarrow n|k)$

    Утверждение о делимости входит в условие задачи на доказательство. Выражение m - константное, но не является десятичной записью числа. Антецедент выделен указателем "идентификатор". Его правая часть обрабатывается вспомогательной задачей на преобразование, имеющей цели "упростить" и "число". В результате получается десятичная константа k. Уровень срабатывания равен 6.

Непосредственное усмотрение делимости

Приемы этого раздела имеют заголовок "второйтерм".

1. Делимость нуля.

   $\forall_n(n - \text{целое} \to n|0)$

   Антецедент обрабатывается проверочным оператором. Уровень срабатывания равен 1. Эти же замечания относятся к трем следующим приемам.

2. Делимость на единицу.

   $\forall_n(n - \text{целое} \to 1|n)$

3. Делимость целого числа на себя.

   $\forall_n(n - \text{целое} \to n|n)$

4. Делимость произведения на сомножитель.

   $\forall_{mn}(n - \text{целое} \to m|mn)$

5. Делимость целочисленных констант.

   $\forall_{abc}(ab = c \to a|c)$

   Переменные a, c идентифицируются с целочисленными константами. Антецедент выделен указателем "программа". Он обращается к процедуре деления целых чисел. Уровень срабатывания равен 1.

6. Делимость четного числа на 2.

   $\forall_a(a - \text{even} \to 2|a)$

   Антецедент обрабатывается проверочным оператором. Уровень срабатывания равен 1.

Исключение квантора

$\forall_{km}(k - \text{целое}\ \&\ m - \text{целое} \to m|k \leftrightarrow \exists_n(k = mn\ \&\ n - \text{целое}))$

Замена выполняется справа налево. Антецеденты обрабатываются проверочными операторами. Если квантор входит в условие задачи на описание, причем либо k содержит неизвестные, либо имеется цель "учетответа", то прием блокируется, чтобы не разрушить параметрическое описание серии решений. Уровень срабатывания равен 0.

$\forall_{mn}(m - \text{целое}\ \&\ n - \text{целое} \to m|n \leftrightarrow \exists_k(n + km = 0\ \&\ k - \text{целое}))$

Аналогично предыдущему.

$\forall_m(m - \text{натуральное} \to \forall_n(n - \text{натуральное} \to m|n) \leftrightarrow m = 1)$

Замена выполняется слева направо. Антецедент обрабатывается проверочным оператором. Уровень срабатывания равен 0.

$\forall_{mnk}(n - \text{натуральное}\ \&\ m - \text{целое}\ \&\ k - \text{целое} \to \exists_x(n|mx + k\ \&\ x - \text{целое}) \leftrightarrow \text{нод}(m,n)|k)$

Антецеденты обрабатываются проверочными операторами. Уровень срабатывания равен 1.

$\forall_{fgA}(f(y) - \text{натуральное} \to \forall_{xy}(x - \text{натуральное}\ \&\ \neg(x = 1)\ \&\ x|f(y)\ \&\ A(y) \to x = g(y)) \leftrightarrow \forall_y(A(y) \to \text{простое}(f(y))\ \&\ g(y) = f(y)))$

Переменные f, g, A функциональные. Антецедент обрабатывается проверочным оператором, причем вводится дополнительная посылка $A(y)$. Уровень срабатывания равен 2.

Попытка использования параметрического описания

$\forall_{km}(m|k \leftrightarrow \exists_n(n - \text{целое}\ \&\ k = mn))$

Прием имеет заголовок "параметризация". Утверждение о делимости является условием задачи на описание, имеющей цель "пример" либо цель "параметризация" (последнее означает, что ответ должен иметь вид явного параметрического описания). Переменная k - неизвестная, не входящая в m. Прием создает вспомогательную задачу, получаемую рассматриваемой заменой условия. Если ее удается решить, то выдается ответ, иначе - продолжается сканирование текущей задачи. Уровень срабатывания равен 5.

$\forall_{mk}(m - \text{целое}\ \&\ k - \text{целое} \to m|k \leftrightarrow \exists_n(n - \text{целое}\ \&\ k = mn))$

Аналогично предыдущему, но не требуется, чтобы k было неизвестной, а требуется лишь, чтобы рассматриваемое условие содержало неизвестные. Антецеденты обрабатываются проверочными операторами. Уровень срабатывания равен 6.

$\forall_{km}(k - \text{целое} \to m|k \leftrightarrow \exists_n(n - \text{целое}\ \&\ k = mn))$

Прием имеет заголовок "второйтерм". Преобразуемое утверждение входит в условие задачи на описание. Переменная k - неизвестная, выражение m - натуральная константа. Преобразованное условие сопровождается комментарием "серия". Антецедент обрабатывается проверочным оператором. Уровень срабатывания равен 4.

Попытка разложения на множители

$\forall_{mnkpq}(m = k \to m^p q|n \leftrightarrow k^p q|n)$

Прием имеет заголовок "второйтерм". Выражение m представляет собой сумму. Антецедент, выделенный указателем "идентификатор", обращается к оператору "видумножение" для разложения этой суммы на множители. Проверяется, что результат k есть произведение либо степень (с точностью до знака). Уровень срабатывания равен 3.

$\forall_{mnkpq}(m = k \to n|m^p q \leftrightarrow n|k^p q)$

Аналогично предыдущему.

Подбор примера

$\forall_{mn}(m - \text{целое} \ \& \ n - \text{целое} \ \& \ 0 < n - m \ \& \ 0 < m \to \neg(n|m))$

Прием имеет заголовок "подборзначений". Консеквент теоремы приема идентифицируется с условием задачи на описание, имеющей цель "пример". Не должно усматриваться, что m неположительно либо что n не превосходит m. Первые два антецедента обрабатываются проверочными операторами. Два последних антецедента выделены указателем "подборзначений". Они замещают рассматриваемое условие во вспомогательной задаче на описание. Если ее удается решить, то выдается ответ, иначе - продолжается сканирование текущей задачи. Уровень срабатывания равен 4.

$\forall_{mnkx}(n - \text{целое} \ \& \ k - \text{целое} \ \& \ m = nk \ \& \ x = n \to x|m)$

Как и в предыдущем случае, прием имеет заголовок "подборзначений". Консеквент идентифицируется с условием задачи на описание Z, имеющей цель "длялюбого". Такая задача возникает из некоторой другой задачи на описание Z', имеющей условием кванторную импликацию K: антецеденты импликации K передаются в посылки задачи Z, консеквент - замещает кванторную импликацию, причем вводится цель (независит $\ldots$), блокирующая зависимость ответа от переменных связывающей приставки. В рассматриваемом приеме третий антецедент идентифицируется с другим условием либо с посылкой задачи, первые два антецедента - обрабатываются проверочными операторами. Проверяется, что выражение n, выбираемое в качестве значения x, не содержит переменных цели (независит $\ldots$). Четвертый антецедент, выделенный указателем "подборзначений", замещает условие делимости во вспомогательной задаче. Уровень срабатывания равен 4.

Разбор случаев для вычета при решении задачи на доказательство

$\forall_{pmx}(p|m \to \exists_i(i \in \{0, \ldots, p-1\} \ \& \ \exists_y(x = py + i \ \& \ y - \text{целое})))$

Прием имеет заголовок "вывод". Указатель "контрольвывода(делит(умножение(x13 x3)x4))" определяет инициализацию приема при усмотрении в условии задачи на доказательство утверждения вида $mc|d$, где m - целочисленная константа. Указатель "контекст(входит(x23 x4))" идентифицирует переменную x как некоторую переменную выражения d. Антецедент, выделенный указателем "программа", выбирает простой делитель p числа m. Указатель "или(фикс(0)фикс(0 2 1))" определяет развертку квантора существования в дизъюнкцию утверждений $\exists_y(x = py \ \& \ y -$ целое$), \ldots, \exists_y(x = py + p - 1 \ \& \ y -$целое). Эта дизъюнкция заносится в посылки задачи и сопровождается комментарием "разборслучаев". Уровень срабатывания равен 6.

Усмотрение ложности целочисленного равенства из соображений делимости

$$\forall_{abcdp}(p|a \ \& \ \neg(p|(d-c)) \ \& \ b - \text{целое} \ \& \ c - \text{целое} \ \& \ d - \text{целое} \to \neg(ab + c = d))$$

Прием имеет заголовок "второйтерм". Равенство $ab + c = d$ входит в условие задачи на описание. Задача должна иметь также условие вида "целое(x)" либо "натуральное(x)", где x - переменная, входящая в текущее условие. Это требование введено для ускорения, чтобы сразу отсекать нецелочисленные задачи. Переменная a идентифицирована с натуральной константой. Первый антецедент выделен указателем "программа". Он перечисляет простые делители числа a. Остальные антецеденты обрабатываются проверочными операторами. Уровень срабатывания равен 2.

$$\forall_{abcdpq}(p = |a| \ \& \ q = d - c \ \& \ \neg(p|q) \ \& \ b - \text{целое} \to \neg(ab + c = d))$$

Равенство входит в посылку задачи на доказательство либо на исследование. Переменные a, d, e идентифицированы с целочисленными константами, b - с переменной. Первые три антецедента выделены указателем "программа", последний - идентифицируется с посылкой. Уровень срабатывания равен 1.

Расшифровка условия делимости, содержащегося в посылках

$$\forall_{mn}(m|n \leftrightarrow \exists_k(k - \text{целое} \ \& \ n = mk))$$

Утверждение о делимости является посылкой задачи, не имеющей типа "преобразовать". Если задача имеет тип "описать", то у нее не должно быть цели "прямойответ". Таким образом, будет разблокировано последующее исключение квантора существования. Уровень срабатывания равен 6.

Наибольший из делителей натурального числа равен этому числу

$$\forall_n(n - \text{натуральное} \to \text{наибольший}(n, \text{set}_m(m - \text{натуральное} \ \& \ m|n)))$$

Прием имеет заголовок "второйтерм", однако указатель "однозначно" совершенно меняет стандартное его поведение. Вместо замены утверждения "наибольший(...)" на константу "истина", реализуется эквивалентная замена по модифицированной в процессе компиляции теореме:

$$\forall_{nx}(n - \text{натуральное} \to (\text{наибольший}(x, \text{set}_m(m - \text{натуральное} \ \& \ m|n)) \leftrightarrow x = n))$$

Антецедент обрабатывается проверочным оператором. Уровень срабатывания приема равен 2.

$$\forall_n(n - \text{целое} \ \& \ \neg(n = 0) \to \text{наибольший}(|n|, \text{set}_m(m - \text{натуральное} \ \& \ m|n)))$$

Аналогично предыдущему. Указатель "однозначно" модифицирует теорему приема следующим образом:

$\forall_{nx}(n-\text{целое}\ \&\ \neg(n=0) \to (\text{наибольший}(x, \text{set}_m(m-\text{натуральное}\ \&\ m|n)) \leftrightarrow x = |n|))$

Усмотрение существования значения неизвестной

Приемы этого раздела имеют заголовок "связка". Переменные связывающей приставки квантора существования идентифицируются с несущественными неизвестными задачи, подкванторные утверждения - со всеми условиями, имеющими вхождения этих неизвестных.

$\forall_k(k - \text{целое} \to \exists_n(n - \text{натуральное}\ \&\ n|k))$

Вместо "$n-$натуральное" под квантором может находиться "$n-$целое". Антецедент обрабатывается проверочным оператором. Уровень срабатывания равен 1.

$\forall_k(k - \text{число} \to \exists_n(n - \text{натуральное}\ \&\ \neg(n|k)) \leftrightarrow \neg(k=0))$

$\forall_k(k - \text{натуральное} \to \exists_n(n - \text{натуральное}\ \&\ \neg(k|n)) \leftrightarrow \neg(k=1))$

$\forall_k(k - \text{целое} \to \exists_n(n - \text{целое}\ \&\ \neg(n=0)\ \&\ n|k))$

$\forall_{mnk}(m-\text{целое}\ \&\ k-\text{целое}\ \&\ n-\text{целое}\ \&\ \text{заимнопросты}(m,n) \to \exists_x(x-\text{целое}\ \&\ n|(mx+k)))$

$\forall_{kn}(k-\text{целое}\ \&\ n-\text{натуральное} \to \exists_m(m-\text{целое}\ \&\ 0 \leq m\ \&\ m \leq n-1\ \&\ n|(m+k)))$

Антецеденты обрабатываются проверочными операторами. Уровень срабатывания равен 1.

$\forall_{mnp}(n - \text{натуральное}\ \&\ m - \text{целое}\ \&\ p - \text{целое} \to \exists_k(k - \text{целое}\ \&\ n|(km+p)) \leftrightarrow \text{нод}(m,n)|p)$

Антецеденты обрабатываются проверочными операторами. Уровень срабатывания равен 2.

$\forall_{knP}(k-\text{целое}\ \&\ n-\text{целое}\ \&\ P(k) \to \exists_x(x-\text{целое}\ \&\ k \leq x\ \&\ x \leq n\ \&\ P(x)) \leftrightarrow k \leq n)$

Переменная P функциональная. Первые два антецедента обрабатываются проверочными операторами, истинность третьего устанавливается с помощью вспомогательной задачи на доказательство. Введен средний ограничитель трудоемкости. Уровень срабатывания равен 4.

Простейшие уравнения с делимостью

$\forall_{mnx}(m - \text{натуральное}\ \&\ n - \text{натуральное} \to (x - \text{натуральное}\ \&\ m|(nx) \leftrightarrow \exists_k(k - \text{натуральное}\ \&\ x = (mk)/\text{нод}(m,n))))$

$\forall_{mnx}(m-\text{натуральное}\ \&\ n-\text{натуральное} \to (x-\text{целое}\ \&\ m|(nx) \leftrightarrow \exists_k(k-\text{натуральное}\ \&\ x = (mk)/\text{нод}(m,n))))$

Прием имеет заголовок "заменаусловия(второйтерм)". Утверждения "$x-$ натуральное" и "$m|(nx)$" идентифицируются с условиями задачи на описание. Выражение x содержит неизвестные, выражения m, n - не содержат. Антецеденты обрабатываются проверочными операторами. Заменяющее утверждение сопровождается комментарием "серия". Уровень срабатывания равен 3.

Отбрасывание первой точки целочисленного промежутка изменения неизвестной из соображений делимости

$\forall_{mnkpx}(m$ – целое & x – целое & $\neg(p|(nx+k))$ & n – целое & k – целое & p – целое & $p|(mn+k) \to m \leq x \leftrightarrow m+1 \leq x)$

Прием имеет заголовок "второйтерм". Заменяемое утверждение идентифицируется с условием задачи на описание. Переменная x - неизвестная, m - выражение без неизвестных. Второй и третий антецеденты идентифицируются с некоторыми другими условиями задачи, прочие антецеденты - обрабатываются проверочными операторами. Уровень срабатывания равен 3.

Усмотрение истинности либо ложности утверждений о делимости с помощью проверочных операторов

$\forall_{ab}(a|b \to a|b)$

$\forall_{ab}(\neg(a|b) \to \neg(a|b))$

Приемы имеют заголовок "второйтерм". Указатель "новый" определяет повторные попытки применения приема при переводе текущего терма задачи в зону внимания. Антецеденты обрабатываются проверочными операторами. В первом случае используется оператор "усмделит", во втором - оператор "усмнеделит". Приемы этих операторов приводятся ниже. Уровень срабатывания равен 2.

Проверочный оператор "усмделит"

1. Делимость нуля.

   $\forall_n(n$ – целое $\to n|0)$

   Антецедент обрабатывается проверочным оператором.

2. Делимость числа на 1.

   $\forall_a(a$ – целое $\to 1|a)$

3. Делимость числа на себя.

   $\forall_a(a$ – целое $\to a|a)$

4. Усмотрение делимости целочисленных констант.

   $\forall_{abc}(ab = c \to a|c)$

   Переменные a, c идентифицируются с целочисленными константами. Антецедент выделен указателем "программа".

5. Усмотрение делимости целочисленного коэффициента.

   $\forall_{abcd}(b = ad \to a|bc)$

   Переменные a, b идентифицируются с целочисленными константами. Антецедент выделен указателем "программа".

6. Отбрасывание минуса.

   $\forall_{ab}(a|b \to a|-b)$

   $\forall_{an}(a|n \to (-a)|n)$

   Антецеденты обрабатываются проверочными операторами. Указатель "спуск" блокирует альтернативные попытки.

7. Делимость сомножителя.

   $\forall_{abc}(a - \text{целое} \ \& \ b - \text{целое} \ \& \ c - \text{целое} \ \& \ a|b \to a|(bc))$

   Антецеденты обрабатываются проверочными операторами. Предшествующие приемы имели уровень срабатывания 1; уровень срабатывания данного приема равен 2.

8. Наибольший общий делитель.

   $\forall_{ab}(\text{нод}(a,b)|b)$

   Уровень срабатывания равен 1.

9. Транзитивность делимости.

   $\forall_{mnk}(m|n \ \& \ n|k \to m|k)$

   Созданы два приема: один из антецедентов берется в посылках, а другой - обрабатывается проверочным оператором. Уровни срабатывания равны 2.

10. Делимость на 2 произведения чисел, имеющих нечетную сумму.

    $\forall_{abnm}(m - \text{целое} \ \& \ n - \text{целое} \ \& \ (a+b)(\text{mod } 2) = 1 \to 2|m(a+n)(b+n))$

    Переменные a, b идентифицируются с константными выражениями. Первые два антецедента обрабатываются проверочными операторами, третий - выделен указателем "идентификатор". Его левая часть обрабатывается нормализаторами общей стандартизации. Уровень срабатывания равен 1.

11. Делимость на 2 четного числа.

    $\forall_{m}(m - \text{even} \to 2|m)$

    Антецедент обрабатывается проверочным оператором "усмчетное". Чтобы не произошло встречное обращение к оператору "усмделит", введен блокирующий комментарий. Уровень срабатывания равен 1.

12. Делимость степеней, имеющих равные основания.

    $\forall_{mnk}(m - \text{натуральное} \ \& \ 0 \leq n - k \to m^k|m^n)$

    Антецеденты обрабатываются проверочными операторами. Уровень срабатывания равен 1.

13. Делимость суммы.

    $\forall_{mnk}(m|n \ \& \ m|k \to m|(n+k))$

    Антецеденты обрабатываются проверочными операторами. Указатель "дистрибразвертка(фикс(0 2))" определяет одновременную обработку всех слагаемых. Уровень срабатывания равен 2.

Проверочный оператор "усмнеделит"

1. Усмотрение неделимости для целочисленных констант.

   $\forall_{abcd}(c = ab + d \to \neg(a|c))$

   Переменные a, c идентифицируются, соответственно, с натуральной и целочисленной константами. Антецедент выделен указателем "программа". Он выполняет деление с остатком. Фильтр "не(заголовок(x4 0))" обеспечивает проверку того, что остаток ненулевой. Уровень срабатывания равен 1.

2. Отбрасывание слагаемых второго выражения, делящихся на первое.

   $\forall_{abc}(a|b \ \& \ \neg(a|c) \to \neg(a|(b+c)))$

   Антецеденты обрабатываются проверочными операторами. Указатель "спуск(1)" блокирует альтернативные попытки после того, как обработан первый антецедент. Уровень срабатывания равен 2.

3. Неделимость произведения на простое число.

   $\forall_{mnk}(\text{простое}(k) \ \& \ m - \text{целое} \ \& \ n - \text{целое} \ \& \ k - \text{целое} \ \& \ \neg(k|m) \ \& \ \neg(k|n) \to \neg(k|(mn)))$

   Антецеденты обрабатываются проверочными операторами. Указатель "спуск" блокирует альтернативные попытки. Уровень срабатывания равен 3.

4. Неделимость степени на простое число.

   $\forall_{mnk}(\text{простое}(m) \ \& \ n - \text{целое} \ \& \ k - \text{целое} \ \& \ 0 \leq k \ \& \ \neg(m|n) \to \neg(m|n^k))$

   Антецеденты обрабатываются проверочными операторами. Уровень срабатывания равен 3.

13.7 Приемы символа "четное"

Утверждение "четное(a)", прорисовываемое формульным редактором в виде "a – even", означает, что a есть четное целое число. Символ "четное" характеризуется справочниками "арность", "предикатныйсимвол". В отличие от символов "целое" и "натуральное", он не задает основного типа объектов. Наиболее часто символ "четное" используется в антецедентах теорем приемов, уточняющих четность либо нечетность числителя и знаменателя дробного показателя степени. Обычно такие антецеденты обрабатываются проверочными операторами "усмчетное", "усмнечетное". Некоторые из приводимых далее приемов были введены не для решения задач из задачников по элементарной алгебре, а для редактирования автоматически создаваемых фильтров приемов ГЕНОЛОГа, связанных с четностью-нечетностью знаменателей и числителей показателей степени.

Общая стандартизация утверждений

1. Условие четности знаменателя суммы дробей, одна из которых имеет нечетный знаменатель.

   $\forall_{ab}(a - \text{rational} \ \& \ b - \text{rational} \ \& \ \neg(\text{знаменатель}(a) - \text{even}) \to \text{знаменатель}(a + b) - \text{even} \leftrightarrow \text{знаменатель}(b) - \text{even})$

Антецеденты обрабатываются проверочными операторами. Уровень срабатывания равен 3.

2. Условие четности числителя суммы двух дробей, одна из которых имеет нечетные знаменатель и числитель, а другая - нечетный знаменатель.

   $\forall_{ab}(a - \text{rational} \ \& \ b - \text{rational} \ \& \ \neg(\text{знаменатель}(a) - \text{even}) \ \& \ \neg(\text{знаменатель}(b) - \text{even}) \ \& \ \neg(\text{числитель}(a) - \text{even}) \rightarrow \text{числитель}(a+b) - \text{even} \leftrightarrow \neg(\text{числитель}(b) - \text{even}))$

   Аналогично предыдущему.

3. Отбрасывание нечетного множителя.

   $\forall_{ab}(a - \text{целое} \ \& \ \neg(a - \text{even}) \ \& \ b - \text{целое} \rightarrow (ab) - \text{even} \leftrightarrow b - \text{even})$

   Антецеденты обрабатываются проверочными операторами. Уровни срабатывания равны 1 и 5.

4. Отбрасывание четного слагаемого.

   $\forall_{ab}(a - \text{even} \rightarrow (a+b) - \text{even} \leftrightarrow b - \text{even})$

   Антецедент обрабатывается проверочным оператором. Уровень срабатывания равен 1.

5. Отбрасывание нечетного слагаемого.

   $\forall_{ab}(a - \text{целое} \ \& \ \neg(a - \text{even}) \rightarrow (a+b) - \text{even} \leftrightarrow \neg(b - \text{even}))$

   Аналогично предыдущему.

6. Отбрасывание минуса.

   $\forall_{a}((-a) - \text{even} \leftrightarrow a - \text{even})$

   Уровень срабатывания равен 0.

7. Исключение квантора.

   $\forall_{a}(\neg(\forall_{n}(n - \text{even} \ \& \ n - \text{целое} \rightarrow a = n)))$

   $\forall_{a}(\neg(\forall_{n}(\neg(n - \text{even}) \ \& \ n - \text{целое} \rightarrow a = n)))$

   $\neg(\forall_{n}(n - \text{целое} \rightarrow \neg(n - \text{even}))$

   $\neg(\forall_{n}(n - \text{целое} \rightarrow n - \text{even})$

   Приемы имеют заголовок "второйтерм". Уровень срабатывания равен 1.

8. Усмотрение нечетного числа из параметрического описания.

   $\forall_{n}(\exists_{m}(n = 2m + 1 \ \& \ m - \text{целое}) \rightarrow \neg(n - \text{even}))$

   Прием имеет заголовок "второйтерм". Антецедент берется в посылках. Уровень срабатывания равен 1.

9. Условное выражение.

   $\forall_{abP}(a - \text{even} \rightarrow \neg((a \text{ при } P, \text{иначе } b) - \text{even}) \leftrightarrow \neg P \ \& \ \neg(b - \text{even}))$

   $\forall_{abP}(b - \text{even} \rightarrow \neg((a \text{ при } P, \text{иначе } b) - \text{even}) \leftrightarrow P \ \& \ \neg(a - \text{even}))$

   Антецедент обрабатывается проверочным оператором. Уровень срабатывания равен 1.

Четность числителя целого четного числа

$\forall_a(a - \text{целое}\ \&\ a - \text{even} \to \text{числитель}(a) - \text{even})$

Прием имеет заголовок "второйтерм". Антецеденты обрабатываются проверочными операторами. Уровень срабатывания равен 1.

Использование нечетности знаменателя при четном числителе

$\forall_{aP}(\neg(\text{знаменатель}(a) - \text{even})\ \&\ P(a)\ \vee\ \text{числитель}(a) - \text{even} \leftrightarrow \neg(\text{знаменатель}(a) - \text{even})\ \&\ (P(a)\ \vee\ \text{числитель}(a) - \text{even}))$

Прием имеет заголовок "второйтерм". Он позволяет вынести из дизъюнкции наружу условие нечетности знаменателя. Переменная P функциональная. Уровень срабатывания равен 1.

Использование условий четности-нечетности для усмотрения отличия от заданной целочисленной константы

$\forall_a(a - \text{целое}\ \&\ \neg(a - \text{even}) \to \neg(a = 0))$

Прием имеет заголовок "второйтерм". Антецеденты обрабатываются проверочными операторами. Введен ускоряющий фильтр, удостоверяющийся в наличии посылки с заголовком "целое" либо "натуральное". Рассматриваемое отрицание равенства должно являться условием задачи на доказательство. Уровень срабатывания приема равен 4.

$\forall_{ab}(a - \text{even} \to \neg(a = b))$

Прием имеет заголовок "второйтерм". Преобразуемое отрицание равенства является условием либо посылкой задачи. Антецедент берется в контексте. Переменная b идентифицируется с нечетной константой. Уровень срабатывания равен 4.

$\forall_{ab}(\neg(a - \text{even}) \to \neg(a = b))$

Аналогично предыдущему, но b идентифицируется с четной константой.

Упрощение выражения с неизвестными, к которому относится требование четности

$\forall_{pq}(p = q \to p - \text{even} \leftrightarrow q - \text{even})$

Преобразуемое утверждение "$p - \text{even}$" входит в условие задачи на описание. Выражение p содержит неизвестные. Антецедент выделен указателем "идентификатор". Его левая часть обрабатывается вспомогательной задачей на упрощение. Проверяется, что результат упрощения q отличен от p. Уровень срабатывания равен 1.

Разбор случаев по четности либо нечетности неизвестной

$\forall_{abcn}(a - \text{целое}\ \&\ b - \text{целое}\ \&\ c - \text{целое}\ \&\ x - \text{целое}\ \&\ \neg(a - \text{even})\ \&\ (ax^n + b)c/d - \text{even} \to \exists_k(k - \text{целое}\ \&\ x = 2k)\ \vee\ \exists_k(k - \text{целое}\ \&\ x = 2k+1))$

$\forall_{abcn}(a - \text{целое}\ \&\ b - \text{целое}\ \&\ c - \text{целое}\ \&\ x - \text{целое}\ \&\ \neg(a - \text{even})\ \&\ \neg((ax^n + b)c/d - \text{even}) \to \exists_k(k - \text{целое}\ \&\ x = 2k)\ \vee\ \exists_k(k - \text{целое}\ \&\ x = 2k+1))$

Приемы имеют заголовок "выводусловия". Последний антецедент идентифицируется с условием задачи на описание. При этом переменная x идентифицируется с неизвестной, n - с натуральной константой. Первые пять антецедентов обрабатываются проверочными операторами. Из контекста не усматриваются ни четность, ни нечетность x. Выводимое условие снабжается комментариями "разборслучаев", "серия". Уровень срабатывания равен 7.

Усмотрение существования значения неизвестной

$$\forall_{an}(\exists_x(x - \text{even} \ \& \ x - \text{натуральное} \ \& \ \forall_i(i \in \{1, \ldots, n\} \to \neg(x = a(i)))))$$

Прием имеет заголовок "связка". Утверждения "x – even, x – натуральное, $\neg(x = a(1)), \ldots, \neg(x = a(n))$" суть все условия задачи на описание, содержащие неизвестную x. Уровень срабатывания равен 3.

Переход к параметрическому описанию

$$\forall_n(n - \text{целое} \to \neg(n - \text{even}) \leftrightarrow \exists_m(m - \text{целое} \ \& \ n = 2m + 1))$$

$$\forall_n(n - \text{even} \leftrightarrow \exists_m(m - \text{целое} \ \& \ n = 2m))$$

Приемы имеют заголовок "второйтерм". Заменяемое утверждение представляет собой условие задачи на описание, имеющей цель "учетответа". Это означает, что имеет место этап редактирования параметрического описания. Переменная n идентифицирована с неизвестной. Преобразованное условие сопровождается комментарием "серия". Уровень срабатывания равен 0. Созданы версии данных приемов, у которых n может идентифицироваться с произвольным выражением, содержащим неизвестные. Их уровень срабатывания равен 8.

$$\forall_n(n - \text{even} \leftrightarrow \exists_m(m - \text{целое} \ \& \ n = 2m))$$

Преобразуемое утверждение является посылкой задачи, не имеющей типа "преобразовать". В случае задачи на описание должна отсутствовать цель "прямойответ". Переменная n идентифицируется с переменной. Для блокировки обратной замены вводится комментарий "кванторнаясвертка". На той же теореме создана еще одна версия приема. Она выполняет указанную замену внутри выражения вида "мощность(класс(...))". Уровень срабатывания обеих версий равен 0.

Усмотрение истинности либо ложности утверждений о четности с помощью проверочных операторов

$$\forall_a(a - \text{even} \to a - \text{even})$$

$$\forall_a(\neg(a - \text{even}) \to \neg(a - \text{even}))$$

Приемы имеют заголовок "второйтерм". Указатель "новый" определяет повторные попытки применения приема при переводе текущего терма задачи в зону внимания. Антецеденты обрабатываются проверочными операторами. В первом случае используется оператор "усмчетное", во втором - оператор "усмнечетное". Приемы этих операторов приводятся ниже. Уровень срабатывания равен 2.

Проверочный оператор "усмчетное"

Четность либо нечетность десятичных констант проверяется оператором ЛОСа "стандследствие", используемым проверочными операторами в их приемах непосредственного усмотрения. На ГЕНОЛОГе реализованы следующие приемы:

1. Четность знаменателя суммы дробей, одна из которых имеет четный, а другая - нечетный знаменатель.

   $\forall_{ab}(\neg(\text{знаменатель}(a) - \text{even})\ \&\ \text{знаменатель}(b) - \text{even} \to \text{знаменатель}(a+b) - \text{even})$

   Антецеденты обрабатываются проверочными операторами. Указатель "спуск(1)" блокирует альтернативные попытки после установления истинности первого антецедента. Уровень срабатывания равен 2.

2. Четность произведения целого числа на четное.

   $\forall_{mn}(m - \text{even}\ \&\ n - \text{целое} \to (mn) - \text{even})$

   Антецеденты обрабатываются проверочными операторами. Уровень срабатывания равен 1.

3. Четность числителя суммы двух дробей, имеющих нечетные числители и знаменатели.

   $\forall_{ab}(\neg(\text{знаменатель}(a) - \text{even})\ \&\ \neg(\text{числитель}(a) - \text{even})\ \&\ \neg(\text{знаменатель}(b) - \text{even})\ \&\ \neg(\text{числитель}(b) - \text{even}) \to \text{числитель}(a+b) - \text{even})$

   Антецеденты обрабатываются проверочными операторами. Указатель "спуск(1 2)" блокирует альтернативные попытки после установления истинности первых двух антецедентов. Уровень срабатывания равен 2.

4. Отбрасывание минуса.

   $\forall_{a}(a - \text{even} \to (-a) - \text{even})$

   Антецедент обрабатывается проверочным оператором. Альтернативные попытки блокируются. Уровень срабатывания равен 1.

5. Сумма четных слагаемых.

   $\forall_{ab}(a - \text{even}\ \&\ b - \text{even} \to (a+b) - \text{even})$

   Антецеденты обрабатываются проверочными операторами. Уровень срабатывания равен 1.

6. Сумма нечетных слагаемых.

   $\forall_{ab}(\neg(a - \text{even})\ \&\ \neg(b - \text{even}) \to (a+b) - \text{even})$

   Аналогично предыдущему.

7. Четность числителя произведения рационального числа с четным числителем на рациональное число с нечетным знаменателем.

   $\forall_{ab}(a - \text{rational}\ \&\ b - \text{rational}\ \&\ \text{числитель}(a) - \text{even}\ \&\ \neg(\text{знаменатель}(b) - \text{even}) \to \text{числитель}(ab) - \text{even})$

   Антецеденты обрабатываются проверочными операторами. Уровень срабатывания равен 2.

8. Четность числителя суммы двух дробей, имеющих четные числители.

   $\forall_{ab}(a - \text{rational} \;\&\; b - \text{rational} \;\&\; \text{числитель}(a) - \text{even} \;\&\; \text{числитель}(b) - \text{even} \rightarrow \text{числитель}(a+b) - \text{even})$

   Аналогично предыдущему.

9. Четность произведения двух целых чисел, имеющих нечетную сумму.

   $\forall_{abmn}(m - \text{целое} \;\&\; n - \text{целое} \;\&\; (a+b)(\text{mod}2) = 1 \rightarrow m(n+a)(n+b) - \text{even})$

   Переменные a, b идентифицируются с константными выражениями. Первые два антецедента обрабатываются проверочными операторами, третий - выделен указателем "идентификатор". Его левая часть обрабатывается нормализаторами общей стандартизации. Уровень срабатывания равен 1.

10. Четность числителя делимого, если знаменатель делителя нечетный, а результат деления имеет четный числитель.

    $\forall_{ab}(a - \text{rational} \;\&\; b - \text{rational} \;\&\; \text{числитель}(a/b) - \text{even} \;\&\; \neg(\text{знаменатель}(b) - \text{even}) \rightarrow \text{числитель}(a) - \text{even})$

    Третий антецедент берется в посылках, остальные - обрабатываются проверочными операторами. Уровень срабатывания равен 1.

11. Сумма числа и его вычета по модулю 2.

    $\forall_{n}(n - \text{целое} \rightarrow (n + n(\text{mod}2)) - \text{even})$

    Антецедент обрабатывается проверочным оператором. Уровень срабатывания равен 2.

Проверочный оператор "усмнечетное"

1. Нечетный числитель дроби с четным знаменателем.

   $\forall_{a}(\text{знаменатель}(a) - \text{even} \rightarrow \neg(\text{числитель}(a) - \text{even}))$

   Антецедент берется в посылках. Выражение a неконстантное. Уровень срабатывания равен 2.

2. Нечетный знаменатель дроби с четным числителем.

   $\forall_{a}(\text{числитель}(a) - \text{even} \rightarrow \neg(\text{знаменатель}(a) - \text{even}))$

   $\forall_{a}(\exists_{n}(\text{числитель}(a) = 2n \;\&\; n - \text{целое}) \rightarrow \neg(\text{знаменатель}(a) - \text{even}))$

   Аналогично предыдущему.

3. Отбрасывание минуса.

   $\forall_{a}(\neg(a - \text{even}) \rightarrow \neg((-a) - \text{even}))$

   Антецедент обрабатывается проверочным оператором. Указатель "спуск" блокирует альтернативные попытки. Уровень срабатывания равен 1.

4. Нечетный знаменатель суммы дробей с нечетными знаменателями.

   $\forall_{ab}(\neg(\text{знаменатель}(a) - \text{even}) \;\&\; \neg(\text{знаменатель}(b) - \text{even}) \rightarrow \neg(\text{знаменатель}(a+b) - \text{even}))$

   Антецеденты обрабатываются проверочными операторами. Указатель "спуск(1)" блокирует альтернативные пoыптки после установления истинности первого антецедента. Уровень срабатывания равен 2.

5. Нечетный знаменатель произведения дробей с нечетными знаменателями.

   $\forall_{ab}(a - \text{rational}\ \&\ b - \text{rational}\ \&\ \neg(\text{знаменатель}(b) - \text{even})\ \&\ \neg(\text{знаменатель}(a) - \text{even}) \to \neg(\text{знаменатель}(ab) - \text{even}))$

   Антецеденты обрабатываются проверочными операторами. Уровень срабатывания равен 2.

6. Нечетный числитель произведения дробей с нечетными числителями.

   $\forall_{ab}(a - \text{rational}\ \&\ b - \text{rational}\ \&\ \neg(\text{числитель}(b) - \text{even})\ \&\ \neg(\text{числитель}(a) - \text{even}) \to \neg(\text{числитель}(ab) - \text{even}))$

   Этот и два следующих приема аналогичны предыдущему.

7. Нечетный знаменатель произведения дроби с четным числителем на дробь с нечетным знаменателем.

   $\forall_{ab}(b - \text{rational}\ \&\ \text{числитель}(b) - \text{even}\ \&\ a - \text{rational}\ \&\ \neg(\text{знаменатель}(a) - \text{even}) \to \neg(\text{знаменатель}(ab) - \text{even}))$

8. Нечетный числитель суммы дробей с четным и нечетным знаменателями.

   $\forall_{ab}(\text{знаменатель}(b) - \text{even}\ \&\ (a+b) - \text{rational}\ \&\ \neg(\text{знаменатель}(a) - \text{even}) \to \neg(\text{числитель}(a+b) - \text{even}))$

9. Нечетный знаменатель суммы дробей с четным числителем и нечтным знаменателем.

   $\forall_{ab}(\text{числитель}(b) - \text{even}\ \&\ b - \text{rational}\ \&\ \neg(\text{знаменатель}(a) - \text{even}) \to \neg(\text{знаменатель}(a+b) - \text{even}))$

   Антецеденты обрабатываются проверочными операторами. Указатель "спуск(1)" блокирует альтернативные попытки после установления истинности первого антецедента. Уровень срабатывания равен 2.

10. Нечетный знаменатель частного дробей с нечетным знаменателем и нечетным числителем.

    $\forall_{ab}(\neg(b = 0)\ \&\ \neg(\text{знаменатель}(a) - \text{even})\ \&\ \neg(\text{числитель}(b) - \text{even})\ \&\ a - \text{rational}\ \&\ b - \text{rational} \to \neg(\text{знаменатель}(a/b) - \text{even}))$

    Антецеденты обрабатываются проверочными опраорами. Уровень срабатывания равен 2.

11. Нечетный числитель частного дробей с нечетным числителем и нечетным знаменателем.

    $\forall_{ab}(\neg(b = 0)\ \&\ \neg(\text{числитель}(a) - \text{even})\ \&\ \neg(\text{знаменатель}(b) - \text{even})\ \&\ a - \text{rational}\ \&\ b - \text{rational} \to \neg(\text{числитель}(a/b) - \text{even}))$

    Аналогично предыдущему.

12. Сумма четного и нечетного слагаемых.

    $\forall_{ab}(a - \text{even}\ \&\ \neg(b - \text{even}) \to \neg((a+b) - \text{even}))$

    Антецеденты обрабатываются проверочными операторами. Указатель "спуск" блокирует альтернативные попытки. Уровень срабатывания равен 2.

13. Степень нечетного числа.

    $\forall_{mn}(\neg(m - \text{even})\ \&\ n - \text{целое}\ \&\ m - \text{целое} \to \neg(m^n - \text{целое}))$

    Антецеденты обрабатываются проверочными операторами. Уровень срабатывания равен 1.

14. Произведение нечетных чисел.

    $\forall_{mn}(m - \text{целое}\ \&\ n - \text{целое}\ \&\ \neg(m - \text{even})\ \&\ \neg(n - \text{even}) \to \neg(mn - \text{even}))$

    Аналогично предыдущему.

Нормализатор "нормчетное"

Нормализатор выполняет простейшую стандартизацию утверждения о четности. Он содержит рассмотренные выше приемы отбрасывания нечетного множителя либо четного слагаемого, а также приемы усмотрения четности либо нечетности константных выражений с помощью операторов "усмчетное", "усмнечетное".

13.8 Приемы символа "вычет"

Выражение "вычет$(a\ b)$" обозначает вычет целого числа a по натуральному модулю b (берется от 0 до $b-1$). Формульный редактор прорисовывает это выражение в виде $a(\text{mod}b)$. Символ "вычет" характеризуется справочниками "арность", "тип", "одз", "типданных". Кроме того, рассматривается натуральный вычет, принимающий значения от 1 до b. Он обозначается "Вычет$(a\ b)$" и прорисовывается формульным редактором как $a(\text{Mod}b)$.

Общая стандартизация выражений

1. Вычет нуля.

   $\forall_n(n - \text{натуральное} \to 0(\text{mod}n) = 0)$

   Антецедент обрабатывается проверочным оператором. Уровень срабатывания равен 0.

2. Вычет единицы.

   $\forall_n(n - \text{натуральное}\ \&\ 0 \leq n - 2 \to 1(\text{mod}n) = 1)$

   Антецеденты обрабатываются проверочными операторами. Уровень срабатывания равен 0.

3. Нахождение вычета для числовых констант.

   $\forall_{kmnp}(m = kn + p \to m(\text{mod}n) = p)$

   Переменные m, n идентифицируются, соответственно, с целочисленной и натуральной десятичными константами. Антецедент выделен указателем "программа" и выполняет деление с остатком. Уровень срабатывания равен 0.

4. Вычет нечетного числа по модулю 2.

   $\forall_n(n - \text{целое}\ \&\ \neg(n - \text{even}) \to n(\text{mod}2) = 1)$

   Первый антецедент обрабатывается проверочным оператором, второй - берется в посылках. Уровень срабатывания равен 1.

5. Усмотрение нулевого вычета.

   $\forall_{ab}(b|a \to a(\mathrm{mod} b) = 0)$

   Антецедент обрабатывается проверочным оператором. Уровень срабатывания равен 2.

6. Приведение основания степени.

   $\forall_{abcnp}(p = a(\mathrm{mod}\, n)\ \&\ n-$натуральное $\&\ b-$целое $\&\ c-$целое $\&\ m-$натуральное $\to (a^m b + c)(\mathrm{mod}\, n) = (p^m b + c)(\mathrm{mod}\, n))$

   Переменные a, n идентифицируются, соответственно, с целочисленной и натуральной константами. Первый антецедент выделен указателем "идентификатор", его правая часть обрабатывается нормализатором "нормвычет". Проверяется, что результат p отличен от a. Остальные антецеденты обрабатываются проверочными операторами. Уровень срабатывания равен 1.

7. Переход от неконстантного выражения по неконстантному модулю к константному.

   $\forall_{abmnkp}(a -$ целое $\&\ b -$ целое $\&\ n -$ целое $\to (a(n+k)^m + b)(\mathrm{mod}\ n+p) = (a(k-p)^m + b)(\mathrm{mod}\ n+p))$

   Переменные k, p идентифицируются с целочисленными константами, переменная m - с натуральной константой, переменная n - с неконстантным выражением. Преобразуемое выражение входит в условие задачи. Антецеденты обрабатываются проверочными операторами. Уровень срабатывания равен 0.

8. Переход от натурального вычета к обычному.

   $\forall_{ab}(a(\mathrm{Mod}\ b) - 1 = (a-1)(\mathrm{mod} b))$

   Уровень срабатывания равен 2.

Общая стандартизация утверждений

1. Усмотрение четности либо нечетности.

   $\forall_a(a(\mathrm{mod}2) = 0 \leftrightarrow a - \mathrm{even})$

   $\forall_a(a(\mathrm{mod}2) = 1 \leftrightarrow \neg(a - \mathrm{even}))$

   $\forall_a(a(\mathrm{mod}2) - 1 = 0 \leftrightarrow \neg(a - \mathrm{even}))$

   Уровень срабатывания равен 0.

2. Равенство вычета дроби нулю.

   $\forall_{abc}(b -$ целое $\to (a/b)(\mathrm{mod}\ c) = 0 \leftrightarrow a(\mathrm{mod}\ bc) = 0)$

   Антецедент обрабатывается проверочным оператором. Уровень срабатывания равен 0.

3. Переход от неравенства с вычетом к отрицанию равенства.

   $\forall_n(n-$натуральное $\&\ 0 \leq n-2\ \&\ i \in \{1, \ldots, n\} \to i-(i(\mathrm{mod}\, n)) < 2 \leftrightarrow \neg(i = n))$

   $\forall_n(n -$ натуральное $\&\ i \in \{1, \ldots, n\} \to 0 \leq i(\mathrm{mod}\ n)) - i \leftrightarrow \neg(i = n))$

   $\forall_n(n -$ натуральное $\&\ i \in \{1, \ldots, n\} \to 0 \leq i(\mathrm{mod}\ n)) - 1 \leftrightarrow \neg(i = n))$

   Антецеденты обрабатываются проверочными операторами. Уровень срабатывания равен 1.

4. Усмотрение равенства вычета единственному оставшемуся неисключенным значению.

   $\forall_{abcn}(l(b) = n-1 \,\&\, \{0,\ldots,n-1\}\backslash\{;b\} = \{c\} \to \neg(a(\mathrm{mod}\, n) \in \{;b\}) \leftrightarrow a(\mathrm{mod}\, n) = c)$

   Переменная b идентифицируется с выражением вида "набор(...)", переменная n - с натуральной константой. Антецеденты выделены указателем "идентификатор". Уровень срабатывания равен 1.

5. Сокращение общего целочисленного множителя.

   $\forall_{abn}(\neg(n = 0) \,\&\, a - \text{целое} \,\&\, b - \text{целое} \,\&\, n - \text{целое} \to (an)(\mathrm{mod}\ bn) = 0 \leftrightarrow a(\mathrm{mod}\ b) = 0)$

   Антецеденты обрабатываются проверочными операторами. Уровень срабатывания равен 0.

6. Отбрасывание минуса в равенстве вычета нулю.

   $\forall_{ab}((-a)(\mathrm{mod}\ b) = 0 \leftrightarrow a(\mathrm{mod}\ b) = 0)$

   Уровень срабатывания равен 0.

7. Усмотрение отрицания равенства вычета заданной константе.

   $\forall_{abcdn}(d = \text{нод}(b,n) \,\&\, \neg(d|a) \to \neg(a = (bx)(\mathrm{mod}\ n)))$

   Прием имеет заголовок "второйтерм". Переменные a, b идентифицируются с целочисленными константами, n - с натуральной константой. Антецеденты выделены указателем "программа". Уровень срабатывания равен 0.

8. Сокращение равенства с вычетом.

   $\forall_{abcnx}(\text{нод}(b,n) = 1 \,\&\, 0 \le a \,\&\, a < n \,\&\, a = (bc)(\mathrm{mod}\ n) \to a = (bx)(\mathrm{mod}\ n) \leftrightarrow x(\mathrm{mod}\ n) = c)$

   $\forall_{abcdnx}(d = \text{нод}(b,n) \,\&\, 1 < d \,\&\, 0 \le a \,\&\, a < n \,\&\, a = (bc)(\mathrm{mod}\ n) \to a = (bx)(\mathrm{mod}\ n) \leftrightarrow \exists_i(i \in \{0,\ldots,d-1\} \,\&\, x(\mathrm{mod}\ n) = c + (ni)/d))$

   В обоих приемах переменные a, b идентифицируются с целочисленными константами, n - с натуральной константой. Антецеденты выделены указателем "программа". Во втором приеме указатель "или(...)" определяет развертку квантора существования в дизъюнкцию равенств. Уровень срабатывания приемов равен 1.

Объединение двух условий равенства вычета нулю

$\forall_{apqr}(a - \text{целое} \,\&\, r = \text{нок}(p,q) \to a(\mathrm{mod}\ p) = 0 \,\&\, a(\mathrm{mod}\ q) = 0 \leftrightarrow a(\mathrm{mod}\ r) = 0)$

Прием имеет заголовок "второйтерм". Переменные p, q идентифицируются с натуральными константами. Первый антецедент обрабатывается проверочным оператором, второй - выделен указателем "программа". Указатель "дизъюнктоперанд" блокирует попытки идентификации конъюнкции как пары антецедентов кванторной импликации. Уровень срабатывания равен 0.

Объединение двух отрицаний равенства вычета данному значению

$\forall_{abckmn}(m=-a \,\&\, n=-c \,\&\, m\in\{0,\ldots,k-1\} \,\&\, n\in\{0,\ldots,k-1\} \to \neg(a+b(\mathrm{mod}\ k)=0) \,\&\, \neg(c+b(\mathrm{mod}\ k)=0) \leftrightarrow \neg(b(\mathrm{mod}\ k)\in\{m,n\}))$

Прием имеет заголовок "второйтерм". Преобразуемая конъюнкция расположена внутри описателя "класс", связывающая приставка которого пересекается с параметрами выражения b. Переменная k идентифицируется с натуральной константой, переменные a, c - с целочисленными константами. Первые два антецедента выделены указателем "программа", последние два - обрабатываются проверочными операторами.

Уравнения с вычетами

$\forall_{abcx}(\neg(a=0)\to(ax+b)(\mathrm{mod}\ c)=0\leftrightarrow\exists_n(n-\text{целое}\ \&\ x=(-b+cn)/a))$

Уравнение является условием задачи на описание. Переменная x идентифицируется с неизвестной. Выражения a, b неизвестных не содержат. Отсутствует условие вида $x\in\{\ldots\}$. Антецедент обрабатывается проверочным оператором. Заменяющее утверждение сопровождается комментарием "серия". Уровень срабатывания равен 3.

$\forall_{mnx}(x(\mathrm{mod}\ n)=m\leftrightarrow\exists_i(i-\text{целое}\ \&\ x=m(\mathrm{mod}\ n)+ni))$

Уравнение является условием задачи на описание. Переменная x идентифицируется с неизвестной, переменные m, n - с целочисленной и натуральной константами. Заменяющее утверждение сопровождается комментарием "серия". Уровень срабатывания равен 1.

Переход к вычетам для исключения квантора

$\forall_{abn}(a\text{—целое}\ \&\ n\text{—натуральное}\to\exists_m(m\text{—целое}\ \&\ b=mn+a)\leftrightarrow b\text{—целое}\ \&\ b(\mathrm{mod}\ n)=a(\mathrm{mod}\ n))$

Преобразуемое утверждение существования расположено под другим квантором либо описателем, связывающими некоторую его свободную переменную. Антецеденты обрабатываются проверочными операторами. Уровень срабатывания равен 3.

Переход от целочисленного параметра к параметру, пробегающему значения вычета по данному модулю

$\forall_{Pn}(n-\text{натуральное}\to\exists_k(k-\text{целое}\ \&\ P(k(\mathrm{mod}\ n)))\leftrightarrow\exists_k(k\in\{0,\ldots,n-1\}\ \&\ P(k)))$

Указатель "контекст(подчинено(x2 теквхожд)вид(x2 вычет(x11 x14)))" инициирует попытку применения приема при усмотрении выражения "$k(\mathrm{mod}\ n)$". Переменная P функциональная. Для идентификации выражения $P(k(\mathrm{mod}\ n))$ используется указатель "новаргумент(x40 x11 фикс)". Он означает, что переменная k должна встречаться в идентифицирующем выражении только в виде $k(\mathrm{mod}\ n)$. Вхождения данного подвыражения определяют шаблон для построения терма $P(k)$. Антецедент обрабатывается проверочным оператором. Уровень срабатывания равен 2.

Нормализатор общей стандартизации "нормвычет"

Нормализатор имеет следующие приемы, дублирующие рассмотренные выше приемы сканирования задачи: вычет нуля, вычет единицы, нахождение вычета для числовых констант; устранение слагаемых, кратных модулю.

13.9 Приемы символа "целаячасть"

Выражение "целаячасть(a)" обозначает целую часть вещественного числа a. Формульный редактор прорисовывает это выражение в виде $[a]$. Символ "целаячасть" характеризуется справочниками "арность", "тип", "одз", "типданных".

Общая стандартизация выражений

1. Обращение к оператору "нормцелаячасть".

   $\forall_{ab}(b = [a] \to [a] = b)$

   Выражение a константное, антецедент выделен указателем "идентификатор". Его правая часть обрабатывается нормализатором общей стандартизации "нормцелаячасть". Последний использует реализованный на ЛОСе прием для определения целой части константного выражения с помощью процедуры "числоценка". Уровень срабатывания равен 1.

2. Целая часть целого числа.

   $\forall_a(a - \text{целое} \to [a] = a)$

   Антецедент обрабатывается проверочным оператором. Уровень срабатывания равен 3.

3. Упрощение условного выражения с целой частью половины.

   $\forall_n(n - \text{целое} \to (n/2 \text{ при } n - \text{even}, \text{иначе } [n/2]) = [n/2])$

   Антецедент обрабатывается проверочным оператором. Уровень срабатывания равен 0.

4. Вынесение целочисленного слагаемого из-под целой части.

   $\forall_{an}(n - \text{целое} \to [a + n] = n + [a])$

   Антецедент обрабатывается проверочным оператором. Уровень срабатывания равен 1.

5. Устранение целой части при известном значении вычета.

   $\forall_{knp}(n(\text{mod } k) + p = 0 \to [n/k] = (n + p)/k)$

   Переменная k идентифицируется с натуральной константой, p - с целочисленной константой. Антецедент берется в контексте. Допускается вырожденное нулевое значение p. Уровень срабатывания равен 1.

6. Целая часть дробного выражения.

   $\forall_{mn}(m - \text{натуральное} \ \& \ n - \text{натуральное} \ \& \ m|n \to [(n-1)/m] = n/m - 1)$

   Антецеденты обрабатываются проверочными операторами. Уровень срабатывания равен 2.

   $\forall_{ab}(0 \leq a \ \& \ 0 < b \ \& \ 0 < b - a \to [a/b] = 0)$

   Антецеденты обрабатываются проверочными операторами. Уровень срабатывания равен 3.

7. Определение целой части с помощью операторов "верхняяоценка" и "нижняяоценка".

   $\forall_{abcd}(b \leq a \ \& \ a \leq c \ \& \ d = [b] \ \& \ 0 < d - c + 1 \rightarrow [a] = d)$

   Первый и второй антецеденты выделены указателем "значения". Они определяют, соответственно, нижнюю и верхнюю оценки b, c выражения a. Третий антецедент выделен указателем "идентификатор". Его правая часть обрабатывается нормализатором "нормцелаячасть". Проверяется, что результат d - целочисленная константа. Наконец, последний антецедент обрабатывается проверочным оператором. Уровень срабатывания равен 4.

   $\forall_{abc}(b \leq a \ \& \ c = [b] \ \& \ 0 < c + 1 - a \rightarrow [a] = c)$

   Прием аналогичен предыдущему, но вместо верхней оценки выражения a берется оно само. Уровень срабатывания равен 3.

   $\forall_{abcd}(b \leq a \ \& \ a \leq c \ \& \ d = [b] \ \& \ c = d + 1 \rightarrow [a] = (c \text{ при } a = c, \text{ иначе } d))$

   Преобразуемое выражение входит в условие задачи на описание и содержит неизвестные. Первые два антецедента определяют нижнюю и верхнюю оценки b, c, третий - вычисляет целую часть d нижней оценки. Четвертый антецедент выделен указателем "идентификатор" - он убеждается в том, что верхняя оценка c ровно на единицу больше числа d. Уровень срабатывания равен 4.

   $\forall_{abcd}(b \leq a \ \& \ a \leq c \ \& \ d = [b] \ \& \ d - c + 1 = 0 \ \& \ 0 < c - a \rightarrow [a] = d)$

   Аналогично предыдущему, но нет ограничений, связанных с задачей на описание. Последний антецедент обрабатывается проверочным оператором. Уровень срабатывания равен 4.

Усмотрение целого числа

$\forall_n(n - [n] = 0 \leftrightarrow n - \text{целое})$

Уровень срабатывания равен 0.

Равенство нулю целой части половины натурального числа

$\forall_n(n - \text{натуральное} \rightarrow [n/2] = 0 \leftrightarrow n = 1)$

Антецедент обрабатывается проверочным оператором. Уровень срабатывания приема равен 1.

Разбор случаев для целой части выражения при нахождении нижней и верхней его оценок

$\forall_{abcdep}(b \leq a \ \& \ a \leq c \ \& \ d = [b] \ \& \ e = [c] \ \& \ p = e - d \ \& \ 0 < p \rightarrow \exists_i(i \in \{0, \ldots, p\} \ \& \ (b \text{ при } i = 0, \text{ иначе } d + i) \leq a \ \& \ (a \leq c \text{ при } i = p \text{ , иначе } a < d + i - 1) \ \& \ [a] = d + i))$

Прием имеет заголовок "выводусловия". Указатель "контрольвывода(целаячасть(x1))" определяет инициализацию попытки его применения при усмотрении выражения $[a]$ в условии задачи на описание. Проверяется, что a содержит неизвестные и что внутри a отсутствуют содержащие неизвестные целые части. Первый и второй антецеденты, выделенные указателями "значения", находят нижнюю и верхнюю оценки b, c выражения a. Третий и четвертый антецеденты, выделенные указателями "идентификатор", упрощают целые части этих оценок с помощью нормализатора "нормцелаячасть". Результаты d, e должны быть целочисленными константами. Пятый антецедент, выделенный указателем "программа", вычисляет разность p данных констант. Проверяется, что p меньше 15. Последний антецедент обрабатывается проверочным оператором. Указатель "или(...)" определяет развертку заменяющего квантора существования в дизъюнкцию утверждений вида $A \leq a \ \& \ a \leq B \ \& \ [a] = C$, где A, B, C - целочисленные константы. Выводимое утверждение сопровождается комментарием "разборслучаев". Уровень срабатывания равен 5.

Разбор случаев для целой части выражения при нахождении нижней либо верхней его оценки

Приемы имеют заголовок "выводусловия".

$\forall_{abc}(a \leq b \ \& \ c = [b] \rightarrow c \leq a \ \& \ a \leq b \ \vee \ a < c)$

Указатель "контрольвывода(целаячасть(x1))" инициирует попытку применения приема при усмотрении выражения $[a]$ в условии задачи на описание. Проверяется, что a содержит неизвестные и что не удается установить нижнюю оценку для a. Первый антецедент, выделенный указателем "значения", определяет верхнюю оценку b. Второй антецедент, выделенный указателем "идентификатор", определяет с помощью нормализатора "нормцелаячасть" численное значение c целой части верхней оценки. Проверяется, что оно не равно b (иначе - см. следующий прием). Выводимая дизъюнкция сопровождается комментарием "разборслучаев". Уровень срабатывания равен 9.

$\forall_{ab}(a \leq b \rightarrow b-1 \leq a \ \& \ a < b \ \vee \ a = b \ \vee \ a < b-1)$

Аналогично предыдущему, но для случая, когда найденная оценка b - целочисленная константа.

$\forall_{abc}(b \leq a \ \& \ c = [b] \rightarrow b \leq a \ \& \ a < c+1 \ \vee \ c+1 \leq a)$

Прием применяется, если верхнюю оценку найти не удается, но зато находится нижняя оценка b. Результат вычисления целой части этой оценки - целочисленная константа c. Уровень срабатывания равен 9.

Решение уравнений с целой частью

$\forall_{ab}([a] = b \leftrightarrow b - \text{целое} \ \& \ b \leq a \ \& \ a < b+1)$

Равенство является условием задачи на описание. Выражение a содержит неизвестные. Уровень срабатывания равен 3.

$\forall_{ab}([\sqrt{a}] = b \leftrightarrow b - \text{целое} \ \& \ 0 \leq b \ \& \ b^2 \leq a \ \& \ a < b^2 + 2b + 1)$

Аналогично предыдущему, но уровень срабатывания равен 2.

Неравенства с целой частью

1. Редактирование ответа, содержащего неравенства с известными целыми частями.

   $\forall_{ab}(a - b + 1 \leq 0 \rightarrow \neg(0 \leq a - [b]))$

Прием имеет заголовок "второйтерм". Неравенство является условием задачи на описание и не содержит неизвестных. Имеет место этап редактирования ответа. Антецедент обрабатывается проверочным оператором. Уровень срабатывания равен 1.

$\forall_{ab}(0 < 1 - a + b \to 0 \leq -1 + [a] - [b] \leftrightarrow b < [a])$

Аналогично предыдущему. Уровень срабатывания равен 2.

2. Решение простейших неравенств.

   $\forall_{ax}(a - \text{целое} \to a \leq [x] \leftrightarrow a \leq x)$

   $\forall_{ax}(a - \text{целое} \to [x] \leq a \leftrightarrow x < a + 1)$

   Неравенство входит в условие задачи на описание. Либо выражение x содержит неизвестные, либо выражение a отлично от неизвестной, либо имеет место этап редактирования параметрического описания. Антецедент обрабатывается проверочным оператором. Уровень срабатывания равен 2.

   Для строгих неравенств созданы аналогичные приемы:

   $\forall_{an}(n - \text{целое} \to n < [a] \leftrightarrow n + 1 \leq a)$

   $\forall_{an}(n - \text{целое} \to [a] < n \leftrightarrow a < n)$

3. Вывод ослабленного неравенства без целой части из решаемого неравенства.

   $\forall_{abc}(a \leq b \ \& \ c = (a \leq b) \to c)$

   $\forall_{abc}(a < b \ \& \ c = (a < b) \to c)$

   Приемы имеют заголовок "выводусловия". Первый антецедент идентифицируется с условием задачи на описание, содержащим целую часть выражения с неизвестными. Второй антецедент выделен указателем "идентификатор". Его правая часть обрабатывается нормализатором "исклцелаячасть". Если при обращении к нормализатору не вводится комментарий "доказать", то нормализатор выдает ослабленную версию неравенства (являющуюся следствием исходного неравенства), в которой целые части, по мере возможности, исключены. Если же при обращении вводится комментарий "доказать", то нормализатор выдает усиленную версию неравенства (исходное неравенство является ее следствием). В данных приемах комментарий "доказать" при обращении не вводится. Проверяется, что ослабленная версия c не содержит целых частей. Для блокировки повторных попыток используется комментарий "исклцелаячасть". Уровень срабатывания равен 5.

4. Попытка доказать усиленное неравенство без целой части.

   $\forall_{ab}(a \leq b \to a \leq b)$

   $\forall_{ab}(a < b \to a < b)$

   Прием имеет заголовок "второйтерм". Неравенство является условием задачи на доказательство и содержит целую часть неконстантного выражения. Истинность антецедента устанавливается вспомогательной задачей на доказательство, причем предварительно применяется нормализатор "исклцелаячасть" с комментарием "доказать", преобразующий неравенство в его усиленную версию. Введен средний ограничитель трудоемкости. Уровень срабатывания равен 4.

5. Склейка неравенства и равенства с применением целой части.

   $\forall_{afx}(f(-a) \to x = -a \ \&\ a -$ целое $\vee\ -a < x \ \&\ x -$ целое $\&\ f(x) \leftrightarrow -[a] \leq x \ \&\ x -$ целое $\&\ f(x))$

   $\forall_{afx}(f(a) \to x = a \ \&\ a -$ целое $\vee\ a < x \ \&\ x -$ целое $\&\ f(x) \leftrightarrow -[-a] \leq x \ \&\ x -$ целое $\&\ f(x))$

   Дизъюнкция входит в условие задачи на описание. Переменная x идентифицируется с переменной, не входящей в a. Переменная f функциональная. Истинность антецедента устанавливается с помощью вспомогательной задачи на доказательство, которой передается дополнительная посылка "$a -$целое". Уровень срабатывания равен 0.

Нормализатор общей стандартизации "нормцелаячасть"

Нормализатор имеет следующие приемы:

1. Вычисление целой части целого числа.

2. Вынесение целочисленного слагаемого из-под целой части.

3. Целая часть дроби, в которой выделяется слагаемое числителя, кратное знаменателю.

   $\forall_{abn}(n -$ целое $\to [(an + b)/a] = n + [b/a])$

   $\forall_{abn}(n -$ целое $\to [-(an + b)/a] = -n + [-b/a])$

   Антецедент обрабатывается проверочным оператором.

4. Целая часть дробей специального вида.

   $\forall_n(n -$ целое $\to [(2n + 1)/4] = [n/2])$

   $\forall_{mn}(m -$ натуральное $\&\ n -$ натуральное $\&\ m|n \to [(n - 1)/m] = n/m - 1)$

   Антецеденты обрабатываются проверочными операторами.

5. Символы бесконечности.

   $[\infty] = \infty$

   $[-\infty] = -\infty$

Нормализатор "исклцелаячасть"

Как уже отмечалось, нормализатор предназначен для усиления либо ослабления неравенства с целыми частями путем отбрасывании этих целых частей. Если задан комментарий "доказать", то предпринимается усиление неравенства. Результатом будет служить неравенство, следствием которого является исходное неравенство. При отсутствии комментария "доказать" происходит ослабление неравенства. Результатом служит неравенство, являющееся следствием исходного.

1. Отбрасывание целой части в слагаемом при ослаблении неравенства.

   $\forall_{abcd}(0 \leq a \to a[b] + c \leq d \leftrightarrow ab - a + c \leq d)$

   $\forall_{abcd}(0 \leq a \to a[b] + c < d \leftrightarrow ab - a + c < d)$

$\forall_{abcd}(a \leq 0 \to a[b] + c \leq d \leftrightarrow ab + c \leq d)$

$\forall_{abcd}(a \leq 0 \to a[b] + c < d \leftrightarrow ab + c < d)$

$\forall_{abcd}(0 \leq a \to d \leq a[b] + c \leftrightarrow d \leq ab + c)$

$\forall_{abcd}(0 \leq a \to d < a[b] + c \leftrightarrow d < ab + c)$

$\forall_{abcd}(a \leq 0 \to d \leq a[b] + c \leftrightarrow d \leq ab - a + c)$

$\forall_{abcd}(a \leq 0 \to d < a[b] + c \leftrightarrow d < ab - a + c)$

Антецедент обрабатывается проверочным оператором. Комментарий "доказать" отсутствует.

2. Отбрасывание целой части в слагаемом числителя при ослаблении неравенства.

$\forall_{abcdef}(0 \leq be \to f < a + (b[c] + d)/e \leftrightarrow f < a + (bc + d)/e)$

$\forall_{abcdef}(0 \leq be \to f \leq a + (b[c] + d)/e \leftrightarrow f \leq a + (bc + d)/e)$

$\forall_{abcdef}(be \leq 0 \to f < a + (b[c] + d)/e \leftrightarrow f < a + (bc - b + d)/e)$

$\forall_{abcdef}(be \leq 0 \to f \leq a + (b[c] + d)/e \leftrightarrow f \leq a + (bc - b + d)/e)$

$\forall_{abcdef}(0 \leq ad \to (a[b] + c)/d + e < f \leftrightarrow (ab - a + c)/d + e < f)$

$\forall_{abcdef}(0 \leq ad \to (a[b] + c)/d + e \leq f \leftrightarrow (ab - a + c)/d + e \leq f)$

$\forall_{abcdef}(ad \leq 0 \to (a[b] + c)/d + e < f \leftrightarrow (ab + c)/d + e < f)$

$\forall_{abcdef}(ad \leq 0 \to (a[b] + c)/d + e \leq f \leftrightarrow (ab + c)/d + e \leq f)$

Аналогично предыдущему.

3. Отбрасывание целой части в слагаемом при усилении неравенства.

$\forall_{abcd}(0 \leq a \to a[b] + c \leq d \leftrightarrow ab + c \leq d)$

$\forall_{abcd}(0 \leq a \to a[b] + c < d \leftrightarrow ab + c < d)$

$\forall_{abcd}(a \leq 0 \to a[b] + c \leq d \leftrightarrow ab - a + c \leq d)$

$\forall_{abcd}(a \leq 0 \to a[b] + c < d \leftrightarrow ab - a + c < d)$

$\forall_{abcd}(0 \leq a \to d \leq a[b] + c \leftrightarrow d \leq ab - a + c)$

$\forall_{abcd}(0 \leq a \to d < a[b] + c \leftrightarrow d < ab - a + c)$

$\forall_{abcd}(a \leq 0 \to d \leq a[b] + c \leftrightarrow d \leq ab + c)$

$\forall_{abcd}(a \leq 0 \to d < a[b] + c \leftrightarrow d < ab + c)$

Антецедент обрабатывается проверочным оператором. Имеется комментарий "доказать".

4. Отбрасывание целой части в слагаемом числителя при усилении неравенства.

$\forall_{abcdef}(0 \leq ad \to f < (a[b] + c)/d + e \leftrightarrow f < (ab - a + c)/d + e)$

$\forall_{abcdef}(0 \leq ad \to f \leq (a[b] + c)/d + e \leftrightarrow f \leq (ab - a + c)/d + e)$

$\forall_{abcdef}(ad \leq 0 \to f < (a[b] + c)/d + e \leftrightarrow f < (ab + c)/d + e)$

$\forall_{abcdef}(ad \leq 0 \to f \leq (a[b] + c)/d + e \leftrightarrow f \leq (ab + c)/d + e)$

$\forall_{abcdef}(0 \leq ad \to (a[b] + c)/d + e < f \leftrightarrow (ab + c)/d + e < f)$

$\forall_{abcdef}(0 \leq ad \to (a[b] + c)/d + e \leq f \leftrightarrow (ab + c)/d + e \leq f)$

$\forall_{abcdef}(ad \leq 0 \to (a[b]+c)/d+e<f \leftrightarrow (ab-a+c)/d+e<f)$

$\forall_{abcdef}(ad \leq 0 \to (a[b]+c)/d+e\leq f \leftrightarrow (ab-a+c)/d+e\leq f)$

Аналогично предыдущему.

13.10 Приемы символов "нод", "нодвсех"

Выражение "нод($a_1 \dots a_n$)" обозначает наибольший общий делитель целых чисел $a_1, \dots, a_n$. Оно прорисовывается формульным редактором так же, как текстовым. Справочники "ассоциативно", "коммутативно", "тип" указывают, что операция "нод" ассоциативна, коммутативна и принимает натуральные значения. Чтобы можно было рассматривать наибольший общий делитель произвольного конечного множества целых чисел a, введена операция "нодвсех(a)". Ее прорисовка формульным редактором, как и для операции "нод", не отличается от прорисовки текстовым редактором.

Общая стандартизация выражений

1. На ЛОСе реализован прием, вычисляющий наибольший общий делитель числовых констант. Уровень срабатывания равен 1.

2. Наибольший общий делитель двух равных чисел.

   $\forall_n(n - \text{целое} \ \& \ \neg(n=0) \to \text{нод}(n,n)=|n|)$

   Антецеденты обрабатываются проверочными операторами. Уровень срабатывания равен 1.

3. Наибольший общий делитель единицы и целого числа.

   $\forall_n(n - \text{целое} \to \text{нод}(1,n)=1)$

   Антецедент обрабатывается проверочным оператором. Уровень срабатывания равен 0.

4. Наибольший общий делитель нуля и ненулевого целого числа.

   $\forall_n(n - \text{целое} \ \& \ \neg(n=0) \to \text{нод}(0,n)=|n|)$

   Аналогично предыдущему.

5. Вынесение наружу общего целочисленного сомножителя.

   $\forall_{mnk}(m - \text{целое} \ \& \ n - \text{целое} \ \& \ k - \text{целое} \ \& \ \neg(m=0) \to \text{нод}(mn,mk) = |m|\text{нод}(n,k))$

   Выражение m отлично от единицы. Антецеденты обрабатываются проверочными операторами. Уровень срабатывания равен 2.

6. Вынесение наружу общего знаменателя.

   $\forall_{mnk}(k|m \ \& \ k|n \ \& \ \neg(k=0) \to \text{нод}(m/k,n/k)=\text{нод}(m,n)/k)$

   Антецеденты обрабатываются проверочными операторами. Уровень срабатывания равен 2.

7. Одно из чисел делится на другое.

   $\forall_{mn}(m|n \mathbin{\&} \neg(m=0) \to \text{нод}(m,n)=|m|)$

   Антецеденты обрабатываются проверочными операторами. Уровень срабатывания равен 1.

8. Взаимно простые числа.

   $\forall_{mn}(\text{взаимнопросты}(m,n) \to \text{нод}(m,n)=1)$

   $\forall_{mnk}(\text{взаимнопросты}(m,n) \mathbin{\&} k - \text{целое} \to \text{нод}(mk,n)=\text{нод}(k,n))$

   Антецеденты обрабатываются проверочными оператором. Уровень срабатывания первого приема равен 2, второго - 3.

9. Деление с остатком целочисленных коэффициентов подобных слагаемых.

   $\forall_{abcdmn}(a - \text{целое} \mathbin{\&} b - \text{целое} \mathbin{\&} c - \text{целое} \mathbin{\&} d=b-kc \mathbin{\&} m=kn+p \mathbin{\&} |n| \leq |m| \to \text{нод}(ma+b,na+c)=\text{нод}(pa+d,na+c))$

   Переменные m,n идентифицируются с целочисленными константами, выражение a неконстантное. Первые три антецедента обрабатываются проверочными операторами. Пятый и шестой антецеденты выделены указателем "программа". Они проверяют, что модуль n не превосходит модуля m и выполняют деление с остатком m на n. Четвертый антецедент, выделенный указателем "идентификатор", определяет остаточные слагаемые d, возникающие при вычитании из первого операнда результата умножения второго операнда на неполное частное k. Требуется, чтобы либо b,c оба были константными, либо d имело меньше слагаемых, чем b. Уровень срабатывания равен 2.

Общая стандартизация утверждений

1. Равенство наибольшего общего делителя одному из чисел.

   $\forall_{mn}(m - \text{целое} \mathbin{\&} n - \text{натуральное} \to \text{нод}(m,n)=n \leftrightarrow n|m)$

   Антецеденты обрабатываются проверочными операторами. Уровень срабатывания равен 1.

2. Переход к новым параметрам в кванторной импликации.

   $\forall_{PQf}(\forall_{ms}(m - \text{целое} \mathbin{\&} Q(m,s) \to P(\text{нод}(m,f(s)))) \leftrightarrow \forall_{kpqs}(k - \text{натуральное} \mathbin{\&} p - \text{целое} \mathbin{\&} q - \text{целое} \mathbin{\&} \text{взаимнопросты} \mathbin{\&} Q(kp,s) \mathbin{\&} f(s)=kq \to P(k)))$

   Замена параметров заключается в том, что выражения $m, f(s)$, для которых в консеквенте кванторной импликации рассматривается наибольший общий делитель, представляются в виде произведений этого делителя k на взаимно простые числа p,q. Переменные f,P,Q функциональные. Указатель "вхождение(x40)" определяет идентификацию шаблона $P(\ldots)$ по какому-либо вхождению в консеквент выражения нод$(m,f(s))$. Уровень срабатывания равен 4.

Простейшие уравнения

$\forall_{mnpx}(\text{простое}(p) \mathbin{\&} m - \text{натуральное} \mathbin{\&} n - \text{целое} \mathbin{\&} 0 \leq n \mathbin{\&} x - \text{целое} \to \text{нод}(x,p^m)=p^n \leftrightarrow \exists_k(k - \text{целое} \mathbin{\&} \neg(p|k) \mathbin{\&} x=p^n k) \mathbin{\&} n \leq m)$

Равенство является условием задачи на описание. Выражение x содержит неизвестные, выражения m, n, p - не содержат. Антецеденты обрабатываются проверочными операторами. Преобразованное условие сопровождается комментарием "серия". Уровень срабатывания равен 2.

$\forall_{knx}(x$ – целое $\to$ нод$(x, n) = k \leftrightarrow k$ – натуральное & $k|n$ & $\exists_m(m$ – целое & взаимнопросты(m, n) & $x = mk))$

Равенство является условием задачи на описание. Выражение x содержит неизвестные, выражения k, n - не содержат. При этом x идентифицируется путем группировки всех содержащих неизвестные операндов выражения нод$(\ldots)$. Антецедент обрабатывается проверочным оператором. Преобразованное условие сопровождается комментарием "серия". Уровень срабатывания равен 4.

Подбор примера

$\forall_{xyA}(x = 1$ & x – натуральное & $A(1, 1) \to A(x,$ нод$(x, y)))$

Прием имеет заголовок "подборзначений". Утверждение $A(x,$ нод$(x, y))$ идентифицируется с условием задачи на описание, имеющей цель "пример". Переменная A - функциональная. Указатель "вхождение(x26)" определяет идентификацию шаблона $A(x, \ldots)$ путем выделения в текущем условии какого-либо вхождения выражения "нод(x, y)". Переменная x идентифицируется с неизвестной задачи. Первый и третий антецеденты выделены указателем "подборзначений". Они замещают рассматриваемое условие во вспомогательной задаче. Второй антецедент обрабатывается проверочным оператором. Указатель "новый" активирует попытки применения приема при каждом возвращении рассматриваемого условия в зону внимания. Уровень срабатывания равен 1.

Существование решения уравнения

$\forall_{abc}(a$ – целое & b – натуральное $\to \exists_c(c$ – натуральное & $b =$ нод$(a, c)) \leftrightarrow b|a)$

Прием имеет заголовок "связка". Переменная c не входит в выражения a, b. Антецеденты обрабатываются проверочными операторами. Уровень срабатывания приема равен 0.

Условие невырожденности наибольшего общего делителя

$\forall_{mn}(m$–целое & n–целое $\to 1 <$ нод$(m, n) \leftrightarrow \exists_k(k$–натуральное & $1 < k$ & $k|m$ & $k|n))$

Антецеденты обрабатываются проверочными операторами. Уровень срабатывания равен 4.

Расшифровка по определению при доказательстве тождества

$\forall_{abc}(a$ – целое & b – целое & c – целое $\to a -$ (нод$(b, c)) = 0 \leftrightarrow 0 \leq a$ & $a|b$ & $a|c$ & $\forall_x(x|b$ & $x|c \to x \leq a))$

Равенство является условием задачи на доказательство. Если оно представляет собой разность двух наибольших общих делителей, то преобразуется тот из них, который задается более длинным выражением. Антецеденты обрабатываются проверочными операторами. Уровень срабатывания равен 6.

Нормализатор общей стандартизации "нормнод"

Кроме реализованного на ЛОСе приема, вычисляющего наибольший общий делитель двух целочисленных констант, нормализатор имеет следующие приемы, дублирующие рассмотренные выше приемы сканирования задачи:

1. Наибольший общий делитель двух равных чисел.
2. Наибольший общий делитель единицы и целого числа.
3. Наибольший общий делитель нуля и ненулевого целого числа.
4. Вынесение наружу целочисленного сомножителя.
5. Вынесение наружу общего знаменателя.
6. Одно из чисел делится на другое.

Нормализатор общей стандартизации "нормнодвсех"

Нормализатор имеет несколько простых приемов для работы с наибольшими общими делителями конечных множеств целых чисел.

1. Одноэлементное множество.

   $\forall_a(a - \text{целое} \to \text{нодвсех}\{a\} = a)$

2. Определение наибольшего общего делителя двух элементов множества.

   $\forall_{amn}(\text{нодвсех}\{m, n; a\} = \text{нодвсех}\{\text{нод}(m, n); a\})$

   Переменные m, n идентифицируются с целочисленными константами. Выражение нод(m, n) обрабатывается нормализатором общей стандартизации "нормнод".

   $\forall_{amn}(k - \text{натуральное} \to \text{нодвсех}\{mk, nk; a\} = \text{нодвсех}\{k\text{нод}(m, n); a\})$

   Переменные m, n идентифицируются с целочисленными константами.

3. Усмотрение взаимно простых чисел.

   $\forall_a(\text{нодвсех}\{1; a\} = 1)$

13.11 Приемы символов "нок", "ноквсех"

Выражение "нок$(a_1 \dots a_n)$" обозначает наименьшее общее кратное целых чисел a_1, $\dots, a_n$. Оно прорисовывается формульным редактором так же, как текстовым. Справочники "ассоциативно", "коммутативно", "тип" указывают, что операция "нок" ассоциативна, коммутативна и принимает натуральные значения. Чтобы можно было рассматривать наименьшее общее кратное произвольного конечного множества целых чисел a, введена операция "ноквсех(a)". Ее прорисовка формульным редактором, как и для операции "нок", не отличается от прорисовки текстовым редактором.

Общая стандартизация выражений

1. На ЛОСе реализован прием, вычисляющий наименьшее общее кратное двух числовых констант. Уровень срабатывания равен 1.

2. Наименьшее общее кратное двух равных чисел.

   $\forall_n(n$ – целое & $\neg(n = 0) \to$ нок$(n, n) = |n|)$

   Антецеденты обрабатываются проверочными операторами. Уровень срабатывания равен 1.

3. Наименьшее общее кратное единицы и целого числа.

   $\forall_m(m$ – целое & $\neg(m = 0) \to$ нок$(1, m) = |m|)$

   Антецеденты обрабатываются проверочными операторами. Уровень срабатывания равен 0.

4. Вынесение общего целочисленного множителя.

   $\forall_{mnk}(m$ – целое & n – целое & k – целое & $\neg(m = 0) \to$ нок$(mn, mk) = |m|$нок$(n, k))$

   Антецеденты обрабатываются проверочными операторами. Уровень срабатывания равен 1.

5. Произведение наименьшего общего кратного и наибольшего общего делителя.

   $\forall_{mn}(m$ – натуральное & n – натуральное $\to$ нод(m, n)нок$(m, n) = mn)$

   Антецеденты обрабатываются проверочными операторами. Уровень срабатывания равен 1.

6. Взаимно простые числа.

   $\forall_{mn}($взаимнопросты$(m, n) \to$ нок$(m, n) = |mn|)$

   Антецедент обрабатывается проверочным оператором. Уровень срабатывания равен 2.

7. Использование выделения степени простого делителя.

   $\forall_{abmnp}(a$ – целое & b – целое & простое(p) & m – целое & n – целое & $0 \leq m$ & $0 \leq n$ & взаимнопросты(p, a) & взаимнопросты$(p, b) \to$ нок$(p^n a, p^m b) = p^{\max(m,n)}$нок$(a, b))$

   Переменная p идентифицируется с натуральной константой. Антецеденты обрабатываются проверочными операторами.Уровень срабатывания равен 4.

8. Использование нормализатора "нормноквсех".

   $\forall_{ab}($ноквсех$(a) = b \to$ ноквсех$(a) = b)$

   Антецедент выделен указателем "идентификатор". Его левая часть обрабатывается нормализатором общей стандартизации "нормноквсех". Проверяется, что результат b не имеет заголовок "ноквсех". Уровень срабатывания приема равен 2.

Общая стандартизация утверждений

1. Равенство наименьшего общего кратного одному из чисел.

   $\forall_{mn}(m$ – натуральное & n – натуральное $\to$ нок$(m,n)=n \leftrightarrow m|n)$

   Антецеденты обрабатываются проверочными операторами. Уровень срабатывания равен 1.

2. Ориентация равенства.

   $\forall_{ab}(b=$ ноквсех$(a) \leftrightarrow$ ноквсех$(a)=b)$

   Прием имеет заголовок "второйтерм". Преобразуемое равенство является посылкой задачи. Выражение b не содержит символа "ноквсех". Прием применяется при отсутствии комментария "ориентацияравенства" к рассматриваемой посылке, причем этот комментарий вводится. После применения приема выражения "ноквсех(a)" будут заменяться в задаче на b. Уровень срабатывания равен 0.

Вывод следствий

1. Равенство наименьшего общего кратного двух неизвестных величин числовой константе.

   $\forall_{abcknpxy}(c=\prod_{i=1}^{n}(p(i))^{k(i)}$ & нок$(a,b)=c \to a=\prod_{i=1}^{n}(p(i))^{x(i)}$ & $b=\prod_{i=1}^{n}(p(i))^{y(i)}$ & $\forall_i(i\in\{1,\ldots,n\}\to k(i)=\max(x(i),y(i)))$ & $\forall_i(i\in\{1,\ldots,n\}\to x(i)$ – целое & $y(i)$ – целое & $0\leq x(i)$ & $0\leq y(i)))$

   Прием имеет заголовок "вывод". Антецедент нок$(a,b)=c$ идентифицируется с посылкой задачи на исследование, имеющей цель "известно". Переменная c при этом идентифицируется с натуральной константой. Выражения a,b содержат неизвестные. Хотя бы одно из этих выражений обладает сомножителем, не имеющим вида степени натурального числа. Первый антецедент выделен указателем "идентификатор". Его левая часть обрабатывается нормализатором "видумножение", снабженным комментарием "натуральное". Это дает представление c в виде произведения натуральных степеней простых чисел. Правая часть выделена указателем "развертка", т.е. p идентифицируется с набором этих чисел, а k - с набором их степеней. При этом n идентифируется с длиной данных наборов. Указатели "переменные(x23 x14)", "переменные(x24 x14)" определяют выбор новых переменных $x(1),\ldots,x(n)$ и $y(1),\ldots,y(n)$, а указатель "новыенеизвестные(x23 x24)" - присоединение их к списку неизвестных задачи. Указатели "развертка" определяют преобразование конечных произведений выводимого утверждения в обычные произведения, а кванторных импликаций - в конъюнкции. Таким образом, фактически выводятся утверждения $a=p(1)^{x(1)}\ldots p(n)^{x(n)}, b=p(1)^{y(1)}\ldots p(n)^{y(n)}, k(1)=\max(x(1),y(1)),\ldots,k(n)=\max(x(n),y(n)), x(1)-$ целое$, y(1)-$ целое$,\ldots,x(n)-$ целое$, y(n)-$ целое$, 0\leq x(1), 0\leq y(1),\ldots,0\leq x(n), 0\leq y(n)$. Уровень срабатывания равен 3.

2. Использование равенства наименьшего общего кратного произведению степеней простых чисел.

   $\forall_{akmnpqA}($ноквсех$(A)=\prod_{i=1}^{n}(p(i))^{q(i)}$ & простое$(p(i))$ & различны(p) & $q(i)$ – натуральное & $k\in\{1,\ldots,n\}\to a\in A$ & m – натуральное & $a=(p(k))^{q(k)}m)$

Прием имеет заголовок "вывод". Его теорема выражает тот очевидный факт, что наибольшая натуральная степень простого делителя наименьшего общего кратного нескольких чисел должна делить хотя бы одно из этих чисел. Указатель "контрольвывода(степень(значение(x15 x11)значение(x16 x11)))" определяет инициализацию попытки применения приема при усмотрении в задаче на описание вхождения выражения $p(k)^{q(k)}$. Проверяется, что все неизвестные задачи являются несущественными. Первый и третий антецеденты идентифицируются с посылками задачи. При этом переменные p, q - обычные. Истинность второго, четвертого и пятого антецедентов устанавливается с помощью вспомогательных задач на доказательство. В каждом случае используется дополнительная посылка "$i \in \{1, \ldots, n\}$". При построении выводимого утверждения выбираются новые переменные a, m. Уровень срабатывания равен 4.

Существование решения уравнения

$\forall_{abc}(a - \text{натуральное} \;\&\; b - \text{натуральное} \rightarrow \exists_c(c - \text{натуральное} \;\&\; b = \text{нок}(a, c)) \leftrightarrow a|b)$

Прием имеет заголовок "связка". Антецеденты обрабатываются проверочными операторами. Уровень срабатывания равен 0.

Расшифровка по определению при доказательстве тождества

$\forall_{abc}(a - \text{целое} \;\&\; b - \text{целое} \;\&\; c - \text{целое} \rightarrow a - \text{нок}(b, c) = 0 \leftrightarrow 0 < a \;\&\; b|a \;\&\; c|a \;\&\; \forall_n(b|n \;\&\; c|n \;\&\; 0 \leq n \rightarrow a \leq n))$

Равенство является условием задачи на доказательство. Если оно имеет вид разности двух наименьших общих кратных, то берется то, которое задается более длинным выражением. Уровень срабатывания равен 6.

$\forall_{aA}(a - \text{натуральное} \rightarrow a - \text{ноквсех}(A) = 0 \leftrightarrow \forall_x(x \in A \rightarrow x|a) \;\&\; \forall_b(b - \text{натуральное} \;\&\; \forall_x(x \in A \rightarrow x|b) \rightarrow a \leq b))$

Аналогично предыдущему. Уровень срабатывания равен 5.

Нормализатор общей стандартизации "нормнок"

Нормализатор имеет следующие приемы, дублирующие одноименные приемы сканирования задачи:

1. Вычисление наименьшего общего кратного двух числовых констант (реализован на ЛОСе).
2. Наименьшее общее кратное двух равных чисел.
3. Наименьшее общее кратное единицы и целого числа.
4. Вынесение общего целочисленного множителя.

Нормализатор общей стандартизации "нормноквсех"

1. Одноэлементное множество.

   $\forall_a(a - \text{целое} \rightarrow \text{ноквсех}\{a\} = a)$

2. Определение наименьшего общего кратного пары числовых констант.

   $\forall_{amn}(\text{ноквсех}\{m, n; a\} = \text{ноквсех}\{\text{нок}(m, n); a\})$

   Переменные m, n идентифицируются с целочисленными константами. Выражение нок(m, n) обрабатывается нормализатором "нормнок".

13.12 Приемы символа "простое"

Утверждение "простое(a)"означает, что a есть простое число. Оно прорисовывается формульным редактором так же, как текстовым. Символ характеризуется справочниками "арность", "предикатныйсимвол", "одз".

Нецелое число не является простым

$\forall_n(\neg(n - \text{целое}) \rightarrow \neg(\text{простое}(n)))$

Прием имеет заголовок "воройтерм". Утверждение "простое(n)" входит в условие задачи на описание, причем n не является переменной. Антецедент обрабатывается проверочным оператором. Уровень срабатывания равен 2.

Делитель простого числа, отличный от единицы

$\forall_{mn}(\text{простое}(n)\ \&\ m|n\ \&\ \neg(m - 1 = 0) \rightarrow m = n)$

Прием имеет заголовок "вывод". Первые два антецедента берутся в посылках задачи, последний - обрабатывается проверочным оператором. Уровень срабатывания приема равен 2.

Число, не превосходящее единицы, не является простым

$\forall_n(0 \leq 1 - n \rightarrow \neg(\text{простое}(n)))$

Прием имеет заголовок "воройтерм". Утверждение "простое(n)" входит в условие задачи на описание, причем n не является переменной. Антецедент обрабатывается проверочным оператором. Уровень срабатывания равен 3.

Параметрическое представление простого числа, большего двух, как нечетного

$\forall_n(\text{простое}(n)\ \&\ 0 \leq n - 3 \rightarrow \exists_m(m - \text{натуральное}\ \&\ n = 2m + 1))$

Прием имеет заголовок "вывод". Первый антецедент является посылкой задачи на доказательство, n - переменная. Условие рассматриваемой задачи содержит подутверждение вида "$2k|m$", причем n входит в m. Второй антецедент обрабатывается проверочным оператором. Уровень срабатывания равен 5.

Попытка разложения на множители

$\forall_{mn}(m = n \to \text{простое}(m) \leftrightarrow \text{простое}(n))$

Прием предпринимает попытку разложить на множители сумму, значением которой должно служить простое число. Если эти множители окажутся целочисленными, то возникнет возможность получить новые уравнения, приравняв все сомножители, кроме одного, плюс-минус единице. Заголовок приема - "второйтерм". Выражение m представляет собой сумму. Антецедент выделен указателем "идентификатор". Его левая часть обрабатывается нормализатором "видумножение". Проверяется, что n имеет заголовок "умножение" либо "степень". Уровень срабатывания равен 3.

Усмотрение непростоты числа, имеющего невырожденное разложение на множители

$\forall_{mn}(n - \text{натуральное} \;\&\; m - \text{натуральное} \;\&\; \neg(n - 1 = 0) \;\&\; \neg(m - 1 = 0) \to \neg(\text{простое}(mn)))$

Прием имеет заголовок "второйтерм". Антецеденты обрабатываются проверочными операторами. Уровень срабатывания равен 1.

Разбор случаев для условия простоты произведения двух целых чисел

$\forall_{mn}(m - \text{целое} \;\&\; n - \text{целое} \to \text{простое}(mn) \leftrightarrow |m| = 1 \;\&\; \text{простое}(|n|) \;\vee\; |n| = 1 \;\&\; \text{простое}(|m|))$

Прием имеет заголовок "второйтерм". Антецеденты обрабатываются проверочными операторами. Уровень срабатывания равен 2.

Разбор случаев для вычета целочисленной неизвестной при наличии условия "простое(a)"

$\forall_{m}(m - \text{целое} \to \exists_n(n - \text{целое} \;\&\; m = 3n) \;\vee\; \exists_n(n - \text{целое} \;\&\; m = 3n + 1) \;\vee\; \exists_n(n - \text{целое} \;\&\; m = 3n + 2))$

Прием имеет заголовок "выводусловия". Указатель "контрольвывода(простое(x1))" инициирует попытку его применения, если в условии задачи на описание усматривается подутверждение "простое(a)", содержащее неизвестную m. Антецедент обрабатывается проверочным оператором. Выводимая дизъюнкция снабжается комментариями "разборслучаев" и "серия". Уровень срабатывания равен 8.

Параметрическое описание делителей числа, разложенного на простые сомножители

$\forall_{pknP}(\text{простое}(i) \;\&\; k(i) - \text{натуральное} \to (m - \text{натуральное} \;\&\; m | \prod_{i=1}^{n}(p(i))^{k(i)} \;\&\; P(m)) \leftrightarrow \exists_a(\text{кортеж}(a, n, \mathbf{Z}) \;\&\; \forall_i(i \in \{1, \ldots, n\} \to 0 \leq a(i) \;\&\; a(i) \leq k(i)) \;\&\; m = \prod_{i=1}^{n}(p(i))^{a(i)} \;\&\; P(m)))$

$\forall_{pknP}(\text{простое}(i) \;\&\; k(i) - \text{натуральное} \to (m - \text{целое} \;\&\; 1 \leq m \;\&\; m | \prod_{i=1}^{n}(p(i))^{k(i)} \;\&\; P(m)) \leftrightarrow \exists_a(\text{кортеж}(a, n, \mathbf{Z}) \;\&\; \forall_i(i \in \{1, \ldots, n\} \to 0 \leq a(i) \;\&\; a(i) \leq k(i)) \;\&\; m = \prod_{i=1}^{n}(p(i))^{a(i)} \;\&\; P(m)))$

Приемы имеют заголовок "второйтерм". Преобразуемая конъюнкция расположена непосредственно под описателем либо под квантором существования. Переменные p, k, P функциональные. Антецеденты обрабатываются проверочными операторами, причем вводится дополнительная посылка $i \in \{1, \dots, n\}$. Уровень срабатывания равен 3.

Условие неделимости произведения степеней простых чисел на заданную степень целого числа, отличного от единицы

$$\forall_{apr}(\text{простое}(p(i))\ \&\ a(i) - \text{целое}\ \&\ 0 \leq a(i)\ \&\ k - \text{натуральное} \rightarrow \forall_n(n - \text{натуральное}\ \&\ n^k | \prod_{i=1}^{r}(p(i))^{a(i)} \rightarrow n - 1 = 0) \leftrightarrow \forall_i(i \in \{1, \dots, r\} \rightarrow a(i) \leq k - 1))$$

Прием имеет заголовок "второйтерм". Переменные a, p функциональные. Антецеденты обрабатываются проверочными операторами, причем вводится дополнительная посылка $i \in \{1, \dots, r\}$. Уровень срабатывания равен 2.

Простые числа в целочисленных уравнениях

1. Равенство степени простого числа произведению целочисленных выражений.

   $$\forall_{abcdenp}(\text{простое}(p)\ \&\ n - \text{целое}\ \&\ 0 \leq n\ \&\ c - b = de\ \&\ d - \text{целое}\ \&\ e - \text{целое} \rightarrow ap^n + b = c \leftrightarrow \exists_{km}(a = km\ \&\ \exists_q(q - \text{целое}\ \&\ kp^q = d\ \&\ mp^{n-q} = e\ \&\ 0 \leq q\ \&\ 0 \leq n - q)))$$

   Преобразуемое равенство является условием задачи на описание. Переменные a, p идентифицируются, соответственно, с целочисленной и натуральной константами. Выражение n неконстантное. Четвертый антецедент выделен указателем "идентификатор". Его левая часть обрабатывается нормализатором "видумножение" и затем идентифицируется с произведением двух содержащих неизвестные выражений d, e. Остальные антецеденты обрабатываются проверочными операторами. Внешний квантор существования в заменяющем утверждении выделен указателем "или(фикс(0 2)фикс(0 2 3 1))". Он разворачивается в дизъюнкцию внутренних кванторов существования по всевозможным представлениям целого числа a в виде произведения двух целых чисел k, m. Эта дизъюнкция сопровождается комментариями "серия", "фильтрсерии", "разборслучаев". Уровень срабатывания равен 4.

2. Условие простоты суммы двух степеней с одинаковыми основаниями.

   $$\forall_{mnp}(0 < p - 1\ \&\ m - \text{целое}\ \&\ n - \text{целое}\ \&\ \text{простое}(p^m + p^n) \rightarrow m \leq 0\ \vee\ n \leq 0)$$

   Прием имеет заголовок "выводусловия". Последний антецедент идентифицируется с условием задачи на описание. При этом p - натуральная константа. Хотя бы одно из выражений m, n содержит неизвестные, и ни для одного из них не усматривается неположительность. Первый антецедент выделен указателем "программа", второй и третий - обрабатываются проверочными операторами. Выводимое утверждение снабжается комментарием "разборслучаев". Уровень срсбатывания равен 4.

3. Усмотрение делимости на 3 из простоты двух чисел, отличающихся на 2.

   $$\forall_{mn}(m - n = 2\ \&\ \text{простое}(m)\ \&\ \text{простое}(n) \rightarrow n = 3\ \vee\ 3|(n + 1))$$

Прием имеет заголовок "выводусловия". Второй и третий антецеденты идентифицируются с условиями задачи на описание. Выражение n содержит неизвестные. Не усматривается, что $n+1$ делится на 3. Первый антецедент выделен указателем "идентификатор". Его левая часть обрабатывается нормализаторами общей стандартизации. Выводимая дизъюнкция снабжается комментарием "разборслучаев". Уровень срабатывания равен 6.

4. Усмотрение существования значения неизвестной.

   $\forall_n(n$ – натуральное $\rightarrow \exists_m($простое(m) & $m|n) \leftrightarrow \neg(n=1))$

   Прием имеет заголовок "связка". Уровень срабатывания равен 1.

Проверочный оператор "усмпростое"

Оператор, кроме общего у всех проверочных операторов обращения к процедуре "стандследствие", имеет всего два приема. Первый реализован на ЛОСе и усматривает простоту натуральной константы. Второй использует кванторную посылку:

$\forall_{anA}(A(n)$ & $\forall_i(A(i)$ – простое$(a(i))) \rightarrow$ простое$(a(n)))$

Здесь переменная A функциональная, a - обычная. Второй антецедент берется в посылках. Истинность первого устанавливается с помощью процедуры "очевидно", использующей проверочные операторы.

Проверочный оператор "усмнепростое"

Здесь проверка непростоты десятичной константы реализуется непосредственно процедурой "стандследствие". Кроме этого, имеется единственный прием, усматривающий непростоту числа, не превосходящего единицы:

$\forall_n(0 \leq 1-n \rightarrow \neg($простое$(n)))$
Антецедент обрабатывается проверочным оператором.

13.13 Приемы символа "взаимнопросты"

Утверждение "взаимнопросты(a,b)" означает, что a,b суть взаимно простые целые числа. Прорисовка формульным редактором совпадает с прорисовкой текстовым редактором. Символ "взаимнопросты" характеризуется справочниками "арность", "коммутативно", "предикатныйсимвол", "одз", "типданных".

Переформулировка условия равенства наибольшего общего делителя единице

$\forall_{mn}($нод$(m,n)=1 \leftrightarrow$ взаимнопросты$(m,n))$

Уровень срабатывания равен 1.

Равенство степеней простого числа, домноженных на взаимно простые с ним числа

$\forall_{abmnp}(a$–целое & b–целое & простое(p) & взаимнопросты(p,a) & взаимнопросты(p,b) & m – целое & n – целое & $0 \leq m$ & $0 \leq n \rightarrow p^n a = p^m b \leftrightarrow m=n$ & $a=b)$

Переменная p идентифицируется с натуральной константой. Антецеденты обрабатываются проверочными операторами. Уровень срабатывания равен 1.

Условие делимости взаимно простых чисел

$\forall_{mn}(\text{взаимнопросты}(m, n) \to m|n \leftrightarrow m = 1)$

Уровень срабатывания равен 1.

Отбрасывание несущественных условий на взаимную простоту в посылках кванторной импликации

$\forall_{fAB}(\forall_{kn}(k - \text{целое} \ \& \ \text{взаимнопросты}(k, f(n)) \ \& \ A(n) \to B(n)) \leftrightarrow \forall_n(A(n) \to B(n)))$

Переменные f, A, B функциональные. Уровень срабатывания равен 2.

Усмотрение взаимной простоты с помощью оператора "усмвзаимнопросты"

$\forall_{ab}(\text{взаимнопросты}(a, b) \to \text{взаимнопросты}(a, b))$

Прием имеет заголовок "второйтерм". Антецедент обрабатывается проверочным оператором "усмвзаимнопросты". Уровень срабатывания равен 2.

Проверочный оператор "усмвзаимнопросты"

Оператор имеет следующие приемы:

1. Взаимная простота двух целочисленных констант усматривается приемом, реализованным на ЛОСе.

2. Исключение произведения.

   $\forall_{mnk}(m - \text{целое} \ \& \ n - \text{целое} \ \& \ \text{взаимнопросты}(m, k) \ \& \ \text{взаимнопросты}(n, k) \to \text{взаимнопросты}(mn, k))$

   Антецеденты обрабатываются проверочными операторами. Указатель "спуск" блокирует альтернативные попытки.

3. Исключение степени.

   $\forall_{mnk}(k - \text{целое} \ \& \ 0 \leq k \ \& \ \text{взаимнопросты}(m, n) \to \text{взаимнопросты}(m^k, n))$

   Антецеденты обрабатываются проверочными операторами. Указатель "спуск(1 2)" блокирует альтернативные попытки после установления истинности первых двух антецедентов.

4. Исключение слагаемого, кратного другому числу.

   $\forall_{mnk}(m - \text{целое} \ \& \ n - \text{целое} \ \& \ k - \text{целое} \ \& \ \text{взаимнопросты}(m, k) \to \text{взаимнопросты}(m, mn + k))$

   Антецеденты обрабатываются проверочными операторами.

5. Взаимная простота с единицей.

   $\forall_m(m - \text{целое} \to \text{взаимнопросты}(m, 1))$

   Антецедент обрабатывается проверочным оператором.

13.14 Приемы символа "дробнаячасть"

Выражение "дробнаячасть(a)" обозначает дробную часть вещественного числа a (не меньше нуля и меньше единицы). Формульный редактор прорисовывает это выражение так же, как текстовый. Символ "дробнаячасть" характеризуется справочниками "арность", "тип".

Устранение дробной части внутри другой дробной части

$\forall_{abn}(n-\text{целое}\rightarrow \text{дробнаячасть}(a+n\text{дробнаячасть}(b))=\text{дробнаячасть}(a+bn))$

Антецедент обрабатывается проверочным оператором. Уровень срабатывания приема равен 1.

Дробная часть числа, принадлежащего $[0,1)$

$\forall_a(0\le a\ \&\ 0<1-a\rightarrow \text{дробнаячасть}(a)=a)$

Антецеденты обрабатываются проверочными операторами. Если a - сумма, то попытка применения приема блокируется. Уровень срабатывания равен 1.

Уравнение с дробной частью

$\forall_{abx}(\text{дробнаячасть}(a+x)=b\leftrightarrow \text{дробнаячасть}(x)=\text{дробнаячасть}(b-a)\ \&\ 0\le b\ \&\ 0<1-b)$

Равенство входит в условие задачи на описание. Выражение x идентифицируется с непустой суммой всех слагаемых, содержащих неизвестные. При этом остаточная сумма a тоже непустая. Выражение b не содержит неизвестных. Уровень срабатывания равен 2.

Переход от вещественного параметра к параметру, пробегающему единичный полуинтервал

$\forall_P(\exists_{xy}(P(\text{дробнаячасть}(x))\ \&\ x-\text{число})\leftrightarrow\exists_{xy}(x\in[0,1)\ \&\ P(x)))$

Переменная P функциональная. После того, как усмотрение квантора существования инициировало попытку применения приема, указатель "контекст(подчинено(x2 теквхожд)вид(x2 дробнаячасть))" определяет идентификацию под этим квантором вхождения выражения "дробнаячасть(x)". Далее указатель "новаргумент(x40 x23 фикс)" обеспечивает проверку того, что под квантором и вне конъюнктивного члена "$x-$число" переменная x встречается только внутри выражения "дробнаячасть(x)". Таким образом находится шаблон $P(\ldots)$. Переменная y идентифицируется с остатком связывающей приставки, который может иметь произвольную длину. Указатель "обобщподст(фикс(0 1))" отменяет проверку невхождения переменных y в $P(\ldots)$. Уровень срабатывания равен 2.

Глава 14

Приемы, связанные с комбинаторными функциями

14.1 Приемы символа "суммавсех"

Выражение "суммавсех(a)" обозначает сумму всех значений вещественнозначной функции a на ее конечной области определения. Формульный редактор рассчитан лишь на те случаи, когда функция a задана через описатель "отображение". Если выражение имеет вид "суммавсех(отображение(i ($i-$целое & $m \leq i$ & $i \leq n$) $A(i)$))", то оно прорисовывается формульным редактором в виде:

$$\sum_{i=m}^{n} A(i).$$

В общем случае выражение "суммавсех(отображение(i $P(i)$ $A(i)$))" прорисовывается так:

$$\sum_{i,P(i)} A(i).$$

Символ "суммавсех" характеризуется справочниками "тип" и "развертка" (последний используется компилятором ГЕНОЛОГа при наличии указателя "развертка", определяющего идентификацию либо выписывание конечной суммы как обычной).

Общая стандартизация

1. Суммирование константного выражения.

$$\forall_f(\sum_{x,f(x)} 0 = 0)$$

Переменная f функциональная, связывающая приставка x может иметь произвольную длину. Уровень срабатывания равен 0.

$$\forall_{af}(\sum_{x,f(x)} a = a\mathrm{card}(\mathrm{set}_x(f(x))))$$

Аналогично предыдущему, но a должно быть отлично от 0.

2. Вынесение константного множителя из-под суммы.

$$\forall_{afgh}(\sum_{x,f(x)} \frac{ag(x)}{h(x)} = a\sum_{x,f(x)} \frac{g(x)}{h(x)})$$

$$\forall_{afgh}(\sum_{x,f(x)} \frac{g(x)}{ah(x)} = \frac{1}{a}\sum_{x,f(x)} \frac{g(x)}{h(x)})$$

Переменные f, g, h функциональные. Уровень срабатывания равен 1.

$$\forall_{abfghp}(a - \text{целое}\ \&\ \neg(b=0) \rightarrow \sum_{x,h(x)} \frac{g(x)b^{f(x)+a}}{p(x)} = b^a \sum_{x,h(x)} \frac{g(x)b^{f(x)}}{p(x)})$$

При рассмотрении рядов прием блокируется. Для этого проверяется отсутствие цели (рядтейлора …), а в случае, когда b идентифицировано с переменной, проверяется, что суммирование не происходит по всем целым значениям x начиная с некоторого x_0. Антецеденты обрабатываются проверочными операторами. Уровень срабатывания равен 1.

$$\forall_{abfghp}(0 < b \rightarrow \sum_{x,h(x)} \frac{p(x)}{g(x)b^{f(x)+a}} = \frac{\sum_{x,h(x)} \frac{p(x)}{g(x)b^{f(x)}}}{b^a})$$

Проверяется отсутствие цели (рядтейлора …). Уровень срабатывания равен 3.

3. Вынесение минуса из-под суммы.

$$\forall_{fg}(\sum_{x,f(x)} -g(x) = -\sum_{x,f(x)} g(x))$$

Созданы две версии приема. Первая применяется, если сумма не расположена внутри утверждения о сходимости ряда - тогда результирующая сумма обрабатывается нормализатором "нормсуммавсех". Вторая версия применяется, если сумма располложена внутри указанного утверждения, и тогда нормализаторы не используются. В обоих случаях уровень срабатывания равен 0. Заметим, что в формальной записи утверждения о сходимости ряда (подробнее см. разделы, связанные с приемами решателя по математическому анализу) используется явным образом указываемая последовательность частичных (конечных) сумм, а не обычная запись с бесконечной суммой.

$$\forall_{mnk}(\sum_{i=m}^{n} \frac{1}{k-i} = -\sum_{i=m}^{k} \frac{1}{i-k})$$

Уровень срабатывания равен 1.

4. Заголовок "плюс" выражения под суммой.

$$\forall_{fgmn}(\sum_{i=m}^{n}(f(i)+g(i)) = \sum_{i=m}^{n} f(i) + \sum_{i=m}^{n} g(i))$$

Преобразование блокируется при анализе сходимости рядов, вычислении суммы ряда, а также при решении задач на программирование. Последние формализуются как задачи на описание, имеющие цель "вычисление". Подробнее о таких задачах будет говориться в разделах, связанных с вычислениями на ГЕНОЛОГе. Уровень срабатывания равен 1.

5. Вынесение слагаемого.

$$\forall_{aPt}(\sum_{x,x=a\ \vee\ P(x)} t(x) = t(a) + \sum_{x,P(x)} t(x))$$

Переменные P, t функциональные. Уровень срабатывания равен 3.

6. Вырожденное суммирование и переход к обычной сумме.

$$\forall_{fx}(\sum_{n=x}^{x} f(n) = f(x))$$

$$\forall_{fx}(\sum_{y,y=x} f(y) = f(x))$$

$$\forall_{f}(\sum_{x,\text{ложь}} f(x) = 0)$$

Переменная f функциональная. Уровень срабатывания равен 0.

$$\forall_{afmn}(n-\text{целое}\ \&\ m-\text{целое}\ \&\ a = m-n\ \&\ 0 < a \to \sum_{k=n}^{m} f(k) = f(n) + \sum_{k=n+1}^{m} f(k))$$

Сумма входит в условие задачи на преобразование. Третий антецедент, выделенный указателем "идентификатор", вычисляет разность a между верхним и нижним пределами суммирования. Проверяется, что эта разность задается десятичной константой. Если сумма имеет свободные переменные, то проверяется, что a меньше 4, иначе - что оно меньше 30. Остальные антецеденты обрабатываются проверочными операторами. Переменная f функциональная. Уровень срабатывания равен 3. Для посылок задачи на исследования создан аналогичный прием с уровнем срабатывания 1. В нем требуется, чтобы $f(k)$ содержало неизвестные.

$$\forall_{fmn}(0 < m - n \to \sum_{i=m}^{n} f(i) = 0)$$

Антецедент обрабатывается проверочным оператором. Прием имеет ускоряющий фильтр, проверяющий, что либо верхний предел константный, либо нижний неконстантный. Уровень срабатывания равен 3. Введена еще одна версия этого приема, срабатывающая на уровне 8 и не имеющая ускоряющего фильтра. Для нее предусмотрен достаточно сильный ограничитель трудоемкости.

$$\forall_{Pn}(\sum_{s,\text{Разбиение}(n,s)} P(s) = \sum_{t,\text{Разбиение}(n,t)} P(t))$$

Суммирование ведется по всем разбиениям натурального числа n в сумму нескольких упорядоченных по невозрастанию натуральных чисел. Переменная n

идентифицируется с натуральной константой, меньшей 6. Указатель "развертка(фикс(0 2))" обеспечивает развертку заменяющей конечной суммы в обычную. Перечисление разбиений выполняется реализованной на ЛОСе процедурой "Разбиения". Уровень срабатывания равен 6.

$$\forall_{fPx}(\sum_{y,y=x,P} f(y) = (f(x) \text{ при } P, \text{ иначе } 0))$$

$$\forall_{fPx}(\sum_{y,y=x,P(y)} f(y) = (f(x) \text{ при } P(x), \text{ иначе } 0))$$

В первом приеме условие P не зависит от переменной суммирования y, во втором - зависит. Уровень срабатывания равен 1.

7. Переобозначение связанной переменной.

$$\forall_{fg}(\sum_{x,f(x)} g(x) = \sum_{y,f(y)} g(y))$$

Прием позволяет получать идентичные конечные суммы, которые вначале отличались связанными переменными. Указатель "контекст(позиция(x1 корень) вид(x1 суммавсех(отображение(x24 значение(x2 x24)значение(x7 x24)))))" определяет идентификацию переменной y путем усмотрения в текущем терме задачи подвыражения $\sum_{y,b(y)} g(y)$. Переменные f, g, b функциональные. Во избежание зацикливания проверяется, что номер переменной y меньше, чем переменной x. Уровень срабатывания равен 0.

8. Суммирование по конечному отрезку целых чисел.

$$\sum_{i,i\in\{m,\dots,n\}} a(i) = \sum_{i=m}^{n} a(i)$$

От описания области суммирования с помощью выражения "номера(...)", определяющего отрезок целых чисел, прием переходит к описанию этой области с помощью двух нестрогих неравенств. Уровень срабатывания равен 0.

9. Приведение подобных членов с конечными суммами.

$$\forall_{abfP}(a\sum_{i,P(i)} f(i) + b\sum_{i,P(i)} f(i) = (a+b)\sum_{i,P(i)} f(i))$$

Сумма входит в условие задачи на преобразование, имеющей цель "упростить". Переменные f, P функциональные. Уровень срабатывания равен 4.

10. Исключение параметрического описания области суммирования.

$$\forall_{fhP}(\sum_{x,\exists_y(x=f(y)\ \&\ P(y))} h(x) = \sum_{y,P(y)} h(f(y)))$$

Переменная y идентифицируется со связывающей приставкой произвольной длины. Переменные f, h, P функциональные. Уровень срабатывания равен 1.

11. Сдвиг области суммирования.

$$\forall_{abfm}(m - \text{целое} \to \sum_{n=a}^{b} f(n) = \sum_{n=a+m}^{b+m} f(n-m))$$

Указатель "контекст(...)" определяет идентификацию десятичной константы m при усмотрении внутри выражения $f(n)$ суммы $n+m$. При этом проверяется, что каждое вхождение переменной n в $f(n)$ расположено в сумме вида $n+k+A$, где k - десятичная константа, не меньшая m. Переменная f функциональная. Рассматриваемая сумма находится в условии задачи на преобразование. Уровень срабатывания равен 2. На той же теореме основан еще один прием, у которого m идентифицируется как выражение произвольного вида. Здесь требуется, чтобы существовало ровно одно вхождение n в $f(n)$, не расположенное в показателе степени вида $(-1)^{n+k}$, причем это вхождение должно быть слагаемым суммы $n+m$. Уровень срабатывания тоже равен 2.

$$\forall_{abcdefg}((b < 0 \ \vee \ 0 < b - a) \ \& \ b = ag + c \to \sum_{n=d}^{e} f(an+b) = \sum_{n=d+g}^{e+g} f(an+c))$$

Указатель "контекст(...)" определяет усмотрение под суммой выражения вида $an+b$, где a - натуральная константа, b - целочисленная. Указатель "новаргумент(x6 x14 извлечение)" определяет идентификацию шаблона $f(\ldots)$ путем такого преобразования выражения под суммой, чтобы переменная n встречалась только внутри подвыражений $an+b$. Преобразование обеспечивается нормализатором "извлечение", которому входная информация передается через комментарий (новаргумент $an+b$ n). Нормализатор играет важную роль при формальном интегрировании и будет описан в разделах, посвященных приемам математического анализа. Антецеденты выделены указателем "программа". Первый из них убеждается в том, что b либо отрицательно, либо больше a. Второй выполняет деление с остатком, так что новый свободный член c оказывается неотрицательным и меньшим a. Прием применяется в условиях задач на преобразование. Уровень срабатывания равен 2.

12. Раскрывание скобок в выражении под суммой.

$$\forall_{afgx}(a = f(x) \to \sum_{x,g(x)} f(x) = \sum_{x,g(x)} a)$$

Сумма входит в условие задачи на преобразование. Переменные f, g функциональные. Выражение под суммой имеет вид либо невырожденного произведения, сомножителем которого является сумма, либо произведения (возможно, вырожденного), сомножителем которого является степень суммы с натуральным показателем от 2 до 5. Антецедент выделен указателем "идентификатор", для раскрывания скобок он использует нормализатор "стандплюс". При разложении в ряд Тейлора прием блокируется. Переменная x идентифицируется со связывающей приставкой произвольной длины. Если $g(x)$ имеет вид "n – целое & $k \leq n$", то уровень срабатывания равен 5, иначе он равен 2.

13. Разбиение на две подсуммы для последующего приведения подобных членов.

$$\forall_{abcdefg}(0 < d - f \to a\sum_{n=c}^{d} g(n) + b\sum_{n=e}^{f} g(n) = a\sum_{n=c}^{f} g(n) + a\sum_{n=f+1}^{d} g(n) + b\sum_{n=e}^{f} g(n))$$

$$\forall_{abcdefg}(0 < e - c \to a\sum_{n=c}^{d} g(n) + b\sum_{n=e}^{f} g(n) = a\sum_{n=c}^{d} g(n) + a\sum_{n=c}^{e-1} g(n) + b\sum_{n=e}^{f} g(n))$$

Первый прием выравнивает верхние пределы сумм, второй - нижние. Предварительно проверяется, что вторая (неизменяемая) сумма имеет общие члены с первой. Антецедент обрабатывается проверочным оператором. Уровень срабатывания равен 1.

14. Преобразование отношения степеней под суммой к виду степени дроби.

$$\forall_{abcdefgh}(\sum_{n=e}^{g} \frac{f(n)a^{n+d}}{h(n)b^{n+c}} = \frac{a^d}{b^c}\sum_{n=e}^{g} \frac{f(n)(a/b)^n}{h(n)})$$

Сумма расположена в условии задачи на преобразование, причем не внутри утверждения о сходимости. Переменные f, h функциональные. Уровень срабатывания равен 2.

$$\forall_{bcfgh}(\sum_{n=e}^{g} \frac{f(n)}{h(n)b^{n+c}} = \frac{1}{b^c}\sum_{n=e}^{g} \frac{f(n)(1/b)^n}{h(n)})$$

Аналогично предыдущему, но уровень срабатывания равен 3.

15. Вынесение вычета наружу.

$$\forall_{fpk}((\sum_{x,f(x)} p(x)\mathrm{mod}k)\ \mathrm{mod}k = (\sum_{x,f(x)} p(x))\mathrm{mod}k)$$

Уровень срабатывания равен 0.

16. Изменение направления суммирования.

$$\forall_{fn}(\sum_{i=0}^{n} f(n-i) = \sum_{i=0}^{n} f(i))$$

Указатель "новаргумент(x6 x9 извлечение)" определяет идентификацию шаблона $f(\ldots)$ путем предварительного преобразования выражения под суммой к такому виду, где переменная i встречается только в подвыражениях $n - i$. Для этого используется нормализатор "извлечение". Уровень срабатывания приема равен 1.

$$\forall_{abfn}(\sum_{i=a}^{n-b} f(n-i) = \sum_{i=b}^{n-a} f(i))$$

Аналогично предыдущему.

$$\forall_{pqnA}(\sum_{i=p}^{q} A(i) = \sum_{i=n-q}^{n-p} A(n-i))$$

Переменная n идентифицируется при усмотрении подвыражения C_{n-i}^k. Проверяется, что отсутствуют подвыражения вида C_{m+i}^s и что в сумме не встречается символ "мощность". Переменная A функциональная. Уровень срабатывания равен 2.

$$\forall_{pnt}(\sum_{i=p}^{n} t(i) = \sum_{i=0}^{n-p} t(n-i))$$

Каждое вхождение переменной i в выражение под суммой расположено внутри некоторого "числа сочетаний", причем существует подвыражение $n-i$. Нижний предел p не равен 0. Уровень срабатывания равен 2.

17. Простейшие случаи перехода от двойной суммы к однократной.

$$\forall_{ak}(k-\text{натуральное} \to \sum_{i=1}^{k}(a(i) \sum_{j,j\in\{1,\ldots,k\},|i-j|\leq 1} a(j)) = \sum_{i=1}^{k}(a(i))^2+2\sum_{i=1}^{k-1} a(i)a(i+1))$$

Переменная a функциональная. Антецедент обрабатывается проверочным оператором. Уровень срабатывания равен 1.

$$\forall_{amnP}(0 \leq j \to \sum_{i,j,i+j\leq m,n\leq i,P(j),i-\text{целое},j-\text{целое}} a(i,j) = \sum_{k=n}^{m} \sum_{l,l\leq m-k,P(l),l-\text{целое}} a(k,l))$$

Переменные m, n идентифицируются с целочисленными константами, переменные a, P функциональные. Внешняя конечная сумма заменяющего выражения разворачивается в обычную сумму. Антецедент обрабатывается проверочным оператором. При этом используются дополнительные посылки "$j-$целое, $P(j)$". Уровень срабатывания равен 3.

$$\forall_{ftP}(\sum_{i,j,i=f(j),P(j)} t(i,j) = \sum_{j,P(j)} t(f(j),j))$$

Указатели "элемент(x9)", "элемент(x10)" позволяют идентифицировать i, j без учета их порядка в связывающей приставке. Переменные f, t, P функциональные. Уровень срабатывания равен 2.

18. Исключение условия нечетности параметра суммирования.

$$\forall_{aP}(\sum_{i,i-\text{целое},\neg(i-\text{even}),P(i)} a(i) = \sum_{i,i-\text{целое} \ \& \ P(2i+1)} a(2i+1))$$

Переменные a, P функциональные. Уровень срабатывания равен 3.

19. Явное разрешение условий на параметр суммирования.

$$\forall_{APmn}(P(i) = (i \in \{n,\ldots,m\}) \to \sum_{i,P(i)} A(i) = \sum_{i=n}^{m} A(i))$$

Сумма расположена в условии задачи на преобразование. Переменные A, P функциональные. Условие $P(i)$ на область суммирования не имеет стандартного вида, т.е. одного из следующих: "i – целое & $a \leq i$ & $i \leq b$", "i – натуральное & $a \leq i$ & $i \leq b$", "i – целое & $a \leq i$". Это условие содержит символ "целое" либо "натуральное". Антецедент выделен указателем "идентификатор". Его левая часть обрабатывается вспомогательной задачей на описание, имеющей неизвестную i. Уровень срабатывания равен 4. Создан еще один прием аналогичного типа, где после разрешения относительно неизвестной i возникает условие принадлежности индекса суммирования некоторому конечному списку:

$$\forall_{APmn}(P(i) = (i \in \{;a\}) \to \sum_{i,P(i)} A(i) = \sum_{i,i\in\{;a\}} A(i))$$

Уровень срабатывания здесь равен 6.

20. Переход от степени минус единицы к условному выражению.

    $\forall_k(k - \text{целое} \to (-1)^k = (1 \text{ при } k - \text{even}, \text{ иначе } -1))$

    Степень минус единицы расположена в условии задачи на преобразование внутри общего члена конечной суммы. Выражение k содержит переменную суммирования. Область суммирования конечная. Общий член не содержит символов "произведениевсех", "суммавсех". При разложении в ряд Тейлора прием блокируется. Антецедент обрабатывается проверочным оператором. Уровень срабатывания равен 4.

21. Расширение области суммирования.

    $$\forall_{amn}(b = a(n+1) \to b + \sum_{i=m}^{n} a(i) = \sum_{i=m}^{n+1} a(i))$$

    $$\forall_{amn}(b = a(m-1) \to b + \sum_{i=m}^{n} a(i) = \sum_{i=m-1}^{n} a(i))$$

    Переменная a функциональная. Антецедент выделен указателем "идентификатор". Его правая часть сквозным образом обрабатывается нормализаторами общей стандартизации. Уровень срабатывания равен 4.

Сумма степеней последовательных натуральных чисел

1. Суммирование последовательных чисел.

   $\forall_{mn}(n - \text{целое} \mathbin{\&} m - \text{целое} \mathbin{\&} 0 \le m - n \to \sum_{k=n}^{m} k = (m - n + 1)(m + n)/2)$

   Антецеденты обрабатываются проверочными операторами. Уровень срабатывания равен 1.

   $\forall_{mn}(n - \text{целое} \mathbin{\&} m - \text{целое} \to \sum_{k=n}^{m} k = ((m - n + 1)(m + n)/2 \text{ при } 0 \le m - n + 1, \text{ иначе } 0))$

   Уровень срабатывания равен 3.

2. Суммирование квадратов.

   $\forall_{mn}(n - \text{целое} \mathbin{\&} m - \text{целое} \mathbin{\&} 0 \le m - n \to \sum_{k=n}^{m} k^2 = (m(m+1)(2m+1) - (n-1)n(2n-1))/6)$

   $\forall_{mn}(n - \text{целое} \mathbin{\&} m - \text{целое} \to \sum_{k=n}^{m} k^2 = (m(m+1)(2m+1) - (n-1)n(2n-1))/6 \text{ при } 0 \le m - n, \text{ иначе } 0)$

   Аналогично предыдущему пункту, уровни срабатывания равны 1 для первого приема и 3 для второго.

3. Суммирование кубов.

   $\forall_{mn}(n - \text{целое} \mathbin{\&} m - \text{целое} \mathbin{\&} 0 \le m - n \to \sum_{k=n}^{m} k^3 = m^2(m+1)^2/4 - n - 1)^2 n^2/4)$

   $\forall_{mn}(n - \text{целое} \mathbin{\&} m - \text{целое} \to \sum_{k=n}^{m} k^3 = m^2(m+1)^2/4 - (n-1)^2 n^2/4 \text{ при } 0 \le m - n, \text{ иначе } 0))$

   Аналогично предыдущему.

4. Суммирование степеней последовательных натуральных чисел, более высоких, чем третья степень.

   $\forall_{mnp}(0 \le p-4 \mathbin{\&} m - \text{целое} \mathbin{\&} n - \text{целое} \mathbin{\&} 0 \le m-n \to \sum_{k=n}^{m} k^p = \sum_{k=1}^{p+1}(p!\text{числобернулли}(p+1-k)(m+1)^k/(k!(p+1-k)!)) - \sum_{k=1}^{p+1}(p!\text{числобернулли}(p+1-k)n^k/(k!(p+1-k)!)))$

   Переменная p идентифицируется с натуральной константой. Указатели "развертка" определяют выписывание конечных сумм заменяющего выражения в виде обычных сумм. Нормализатор "нормчислобернулли", описываемый в последующих разделах, вычисляет необходимые числа Бернулли. Антецеденты обрабатываются проверочными операторами. Прием применяется в условиях задач на преобразование. Уровень срабатывания равен 3. Если не усматривается, что верхний предел суммирования не меньше нижнего, используется условное заменяющее выражение:

   $\forall_{mnp}(0 \le p-4 \mathbin{\&} m - \text{целое} \mathbin{\&} n - \text{целое} \to \sum_{k=n}^{m} k^p = (\sum_{k=1}^{p+1}(p!\text{числобернулли}(p+1-k)(m+1)^k/(k!(p+1-k)!)) - \sum_{k=1}^{p+1}(p!\text{числобернулли}(p+1-k)n^k/(k!(p+1-k)!)) \text{ при } 0 \le m-n, \text{ иначе } 0)))$

   Уровень срабатывания здесь тоже равен 3. Чтобы сначала предпринималась попытка применить первый прием, он компилируется до второго. Тогда компилятор "уложит" программы приемов так, что ответвление ко второму произойдет из цепочки операторов первого.

5. Суммирование знакочередующихся степеней последовательных натуральных чисел, более высоких, чем первая степень.

   $\forall_{lmnp}(0 \le p-2 \mathbin{\&} l - \text{целое} \mathbin{\&} n - \text{целое} \mathbin{\&} m - \text{целое} \mathbin{\&} 0 \le n-m \to \sum_{k=m}^{n}((-1)^{k+l}k^p) = (-1)^{l+n}\sum_{i=1}^{p+1}((p!(1-2^i)\text{числобернулли}(i)/(i!(p+1-i)!)) \cdot ((n+1)^{p+1-i} + (-1)^{m+n}m^{p+1-i})))$

   Аналогично предыдущему пункту.

Сумма геометрической прогрессии и сумма, извлекаемая из нее почленным дифференцированием

$\forall_{abckn}(\neg(a-1=0)$ & $k-$целое & $n-$целое & $0 \leq n-k+1$ & $\neg(b=0) \rightarrow \sum_{m=k}^{n} a^{bm/c} = (a^{b(n+1)/c} - a^{bk/c})/(a^{b/c}-1)$

Антецеденты обрабатываются проверочными операторами. Если сумма расположена под квантором общности, связывающего какие-либо ее свободные параметры, то прием блокируется. Уровень срабатывания равен 1.

$\forall_{abkn}(n-$целое & $k-$целое & $0 \leq k-n \rightarrow \sum_{m=n}^{k} a^{bm} = (k-n+1$ при($b=0 \vee a-1=0)$, иначе $(a^{b(k+1)}-a^{bn})/(a^{b}-1)))$

Сумма расположена в условии задачи на преобразование либо содержит неизвестные и расположена в условии задачи на описание. Антецеденты обрабатываются проверочными операторами. Уровень срабатывания равен 2.

$\forall_{aklmn}(\neg(a-1=0)$ & $n-$целое & $l-$целое & $m-$целое & $k-$целое & $0 \leq k-m+1$ & $0 \leq l+m \rightarrow \sum_{p=m}^{k}((p+n)a^{p+l}) = (a^{l}(a^{k+1}(n+k)-(n+m)a^{m})/(a-1)-a^{l+1}(a^{k}-a^{m})/(a-1)^{2}))$

Антецеденты обрабатываются проверочными операторами. Если сумма расположена под квантором общности, связывающего какие-либо ее свободные параметры, либо внутри утверждения о сходимости, то прием блокируется. Уровень срабатывания равен 2.

Преобразование дроби под суммой к виду суммы простейших дробей

$$\forall_{fghp}(p(x) = \frac{g(x)}{h(x)} \rightarrow \sum_{x,f(x)} \frac{g(x)}{h(x)} = \sum_{x,f(x)} p(x))$$

Сумма входит в условие задачи на преобразование. Числитель и знаменатель дроби после раскрывания скобок становятся многочленами от x. Антецедент выделен указателем "идентификатор". Его правая часть обрабатывается нормализатором "простейшиедроби" (см. приемы символа "дробь"), причем в качестве входной переменной берется x. Нормализатор использует дополнительные посылки "число(x)", "$f(x)$". Проверяется, что результат $p(x)$ имеет заголовок "плюс". Уровень срабатывания равен 2.

$$\forall_{fghpq}(\frac{g(i)}{h(i)} = q(i) \rightarrow \sum_{i,p(i)} \frac{f(i)g(i)}{h(i)} = \sum_{i,p(i)} f(i)q(i))$$

Указатель "перечень(x7 контекст(видмногочлена(x7 x9 x1)))" определяет группировку в произведение $g(i)$ всех сомножителей числителя, которые после раскрывания скобок оказываются многочленами от i. Знаменатель тоже должен после раскрывания скобок становиться многочленом от i. При разложении в ряд Тейлора либо в ряд Фурье преобразование блокируется. В остальном прием аналогичен предыдущему. Уровень срабатывания равен 3.

Тригонометрические суммы

1. Суммирование синусов либо косинусов членов арифметической прогрессии.

   $\forall_{abckn}(k-$целое & $n-$целое & $0 \leq n-k+1$ & $\neg(\sin(a/2b)=0) \to \sum_{i=k}^{n} \sin(ia/b+c) = (\cos((2k-1)a/2b+c) - \cos((2n+1)a/2b+c))/2\sin(a/2b))$

   $\forall_{abckn}(k-$целое & $n-$целое & $0 \leq n-k+1$ & $\neg(\sin(a/2b)=0) \to \sum_{i=k}^{n} \cos(ia/b+c) = (\sin((2n+1)a/2b+c) - \sin((2k-1)a/2b+c))/2\sin(a/2b))$

   Выражения a, b, c, k, n не содержат неизвестных. Уровень срабатывания приема равен 1.

2. Преобразование степени минус единицы к виду тригонометрического множителя.

   $$\forall_{abkn}(\sum_{i=k}^{n}((-1)^i \sin(ai/b)) = \sum_{i=k}^{n}(\cos(\pi i)\sin(ai/b)))$$

   $$\forall_{abkn}(\sum_{i=k}^{n}((-1)^i \cos(ai/b)) = \sum_{i=k}^{n}(\cos(\pi i)\cos(ai/b)))$$

   Выражение в заменяющей сумме обрабатывается нормализатором "стандплюс", снабженным комментарием "видумножение". Это обеспечивает преобразование произведений тригонометрических функций в суммы. Уровень срабатывания равен 1.

3. Преобразование произведения тригонометрических функций в сумму.

   $$\forall_{fpkmn}(g(i)(\cos f(i))^k = p(i) \to \sum_{i=m}^{n}(g(i)(\cos f(i))^k) = \sum_{i=m}^{n} p(i))$$

   $$\forall_{fpkmn}(g(i)(\sin f(i))^k = p(i) \to \sum_{i=m}^{n}(g(i)(\sin f(i))^k) = \sum_{i=m}^{n} p(i))$$

   Переменная k идентифицируется с натуральной константой от 1 до 5. Если эта константа равна 1, то $g(i)$ должно иметь своим сомножителем синус либо косинус, зависящий от i. Антецедент выделен указателем "идентификатор". Его правая часть обрабатывается нормализатором "стандплюс", снабженным комментарием "видумножение". Уровень срабатывания равен 3.

Усмотрение фрагмента знакочередующегося гармонического ряда

$\forall_{amnpq}(0 \leq n-m$ & $q = n-p \to 2a\sum_{i=m}^{n} 1/(2i+1) - a\sum_{i=m}^{p} 1/i = 2a\sum_{i=2m}^{2n+1}(-1)^{i+1}/i + (a\sum_{i=p+1}^{n} 1/i$ при $0 < q,$ иначе $- a\sum_{i=n+1}^{p} 1/i))$

Разность входит в условие задачи на преобразование. Первый антецедент обрабатывается проверочным оператором, второй - выделен указателем "идентификатор". Выражение q должно представлять собой целочисленную константу. Уровень срабатывания равен 3.

Суммы с числами сочетаний

1. Усмотрение биномиальной суммы.

   $\forall_{ablmn}(0 \leq n \ \& \ m \leq 0 \ \& \ n - \text{целое} \ \& \ m - \text{целое} \ \& \ l - \text{натуральное} \ \& \ 0 \leq l - n + m \to \sum_{k=n}^{l+m}(a^k b^{l-k} C_l^k) = (a+b)^l - \sum_{k=0}^{n-1}(a^k b^{l-k} C_l^k) - \sum_{k=l+m+1}^{l}(a^k b^{l-k} C_l^k))$

   Переменные m, n идентифицируются с десятичными константами. При этом $n < 4, -4 < m$. Множители a^k, b^{l-k} могут отсутствовать, и тогда a, b идентифицируются с единицей. Антецеденты обрабатываются проверочными операторами. Уровень срабатывания равен 1.

   $\forall_{ablmn}(0 \leq n \ \& \ m \leq 0 \ \& \ n - \text{целое} \ \& \ m - \text{целое} \ \& \ l - \text{целое} \ \& \ 0 \leq l - n + m \to \sum_{k=n}^{l+m}(a^k b^{l-k} C_l^k) = (1 \text{ при } l = 0, \ \text{иначе } (a+b)^l - \sum_{k=0}^{n-1}(a^k b^{l-k} C_l^k) - \sum_{k=l+m+1}^{l}(a^k b^{l-k} C_l^k)))$

   Аналогично предыдущему, но уровень срабатывания равен 2.

   $\forall_{abm}(0 \leq b \ \& \ a \leq 0 \ \& \ m - \text{целое} \ \& \ k - m = a \to \sum_{n=b}^{k} C_m^n = (2^m - \sum_{n=0}^{b-1} C_m^n - \sum_{n=m+a+1}^{m} C_m^n \text{ при } 0 \leq m + a - b, \text{иначе } 0))$

   Первые три антецедента обрабатываются проверочными операторами, последний выделен указателем "идентификатор". Выражения a, b суть десятичные константы, причем $-4 < a, b < 4$. Уровень срабатывания равен 0.

   $\forall_{mnpqk}(n - m = k \ \& \ 0 \leq k \to \sum_{i=0}^{m}(p^i q^{n-i} C_n^i) = (p+q)^n - \sum_{j=m+1}^{n}(p^j q^{n-j} C_n^j))$

   Переменные m, n идентифицируются с натуральными константами. Антецеденты выделены указателем "программа", так что k - целочисленная константа. Проверяется, что $k < 4$. Конечная сумма в заменяющем терме разворачивается в обычную сумму. Уровень срабатывания равен 2.

   $\forall_{afgmpqP}(g(j) + f(j) = m \to \sum_{i,j,i+f(j) \leq m, 0 \leq i, P(j)}((a(j))^i p(j)/i!(g(j) - i)!q(j)) = \sum_{j,P(j),f(j) \leq m}(p(j)(a(j)+1)^{m-f(j)}/q(j)(m - f(j))!))$

   Переменные a, f, g, p, q, P функциональные. Переменная j идентифицируется с остатком связывающей приставки, имеющим произвольную длину. Антецедент выделен указателем "идентификатор". Уровень срабатывания равен 2.

   $\forall_{klmnp}(\sum_{k=n}^{m}(a^{k+p} b^{l-k+q} f(k) C_l^k) = a^p b^q \sum_{k=n}^{m}(a^k b^{l-k} f(k) C_l^k))$

   Переменная f функциональная. Выражения p, q одновременно не равны 0. Уровень срабатывания равен 1.

2. Теорема сложения.

   $\forall_{mnk}(m - \text{целое} \ \& \ n - \text{целое} \ \& \ k - \text{целое} \to \sum_{i=0}^{k}(C_n^i C_m^{k-i}) = C_{m+n}^k)$

   Антецеденты обрабатываются проверочными операторами. Уровень срабатывания равен 1.

   $\forall_{mnk}(m - \text{целое} \ \& \ n - \text{целое} \ \& \ k - \text{целое} \ \& \ p = m - k \to \sum_{i=0}^{p}(C_n^i C_m^{i+k}) = C_{m+n}^p)$

   Первые три антецедента обрабатываются проверочными операторами, последний - выделен указателем "идентификатор". Уровень срабатывания равен 1.

3. Отбрасывание из области суммирования значений, где число сочетаний равно нулю.

   $\forall_{mnkA}(0 \leq k - n \to \sum_{i=0}^{k}(A(i) C_m^{n-i}) = \sum_{i=0}^{n}(A(i) C_m^{n-i}))$

Переменная A функциональная. Антецедент обрабатывается проверочным оператором, причем введен сильный ограничитель трудоемкости. Выражения k, n не должны совпадать. Уровень срабатывания равен 0.

$\forall_{mnkA}(0 \le m - n \to \sum_{i=k}^{m}(A(i)C_n^i) = \sum_{i=k}^{n}(A(i)C_n^i))$

Аналогично предыдущему, но уровень срабатывания равен 1.

4. Сумма произведений чисел сочетаний на натуральные степени чисел, по которым берутся сочетания.

   $\forall_n(n - \text{натуральное} \to \sum_{i=0}^{n}(i^2C_n^i) = n(n+1)2^{n-2})$

   $\forall_n(n - \text{натуральное} \to \sum_{i=0}^{n}(i^3C_n^i) = n^2(n+3)2^{n-3})$

   Уровень срабатывания равен 1.

   $\forall_{pqn}(n - \text{натуральное} \to \sum_{i=0}^{n}(ip^iq^{n-i}C_n^i) = pn(p+q)^{n-1})$

   $\forall_{pqn}(n - \text{натуральное} \to \sum_{i=0}^{n}(i^2p^iq^{n-i}C_n^i) = np(pn+q)(p+q)^{n-2})$

   Указатель "постановка(...)" разрешает идентификацию переменной q с единицей. Уровень срабатывания равен 1.

   $\forall_{abn}(a \le 0 \ \& \ 0 \le b \ \& \ n - \text{натуральное} \ \& \ k - n = a \to \sum_{i=b}^{k}(iC_n^i) = (n2^{n-1} - \sum_{i=0}^{b-1}(iC_n^i) - \sum_{i=0}^{-a-1}((n-i)C_n^i) \text{ при } 0 \le n + a - b, \text{ иначе } 0))$

   Переменные a, b идентифицируются с десятичными константами, причем выполнены неравенства $-4 < a, b < 4$. Первые два антецедента выделены указателем "программа", третий - обрабатывается проверочным оператором, последний - выделен указателем "идентификатор". Уровень срабатывания равен 1.

5. Суммирование чисел сочетаний с фиксированным верхним индексом.

   $\forall_{mnpq}(\sum_{i=p}^{q} C_{n+i}^{m} = C_{n+q+1}^{m+1} - (C_{n+p}^{m+1} \text{ при } 0 \le n + p - m - 1, \text{ иначе } 0))$

   Уровень срабатывания равен 0.

   $\forall_{mnpq}(\sum_{i=p}^{q}(iC_{m+i}^{n}) = (n+1)\sum_{i=p}^{q} C_{m+i+1}^{n+1} - (m+1)\sum_{i=p}^{q} C_{m+i}^{n})$

   Уровень срабатывания равен 2.

   $\forall_{amnkpq}(a(i) = \prod_{j=1}^{k}(m+i+j) - i^k \to \sum_{i=p}^{q}(i^kC_{m+i}^{n}) = \prod_{j=1}^{k}(n+j)\sum_{i=p}^{q} C_{m+k+i}^{n+k} - \sum_{i=p}^{q}(a(i)C_{m+i}^{n}))$

   Переменная k идентифицируется с натуральной константой. Антецедент выделен указателем "идентификатор". Конечное произведение из его правой части разворачивается в обычное произведение, после чего правая часть обрабатывается нормализатором раскрывания скобок "стандплюс". Переменная a функциональная. Конечное произведение из заменяющего терма тоже разворачивается в обычное произведение. Конечные суммы из заменяющего терма обрабатываются вспомогательными задачами на упрощение. Уровень срабатывания приема равен 3.

6. Сумма частных от деления чисел сочетаний на последовательные натуральные числа.

   $\forall_{abn}(0 \le a \ \& \ b \le 0 \to \sum_{i=a}^{n+b}(C_n^i/(i+1)) = (2^{n+1}-1)/(n+1) - \sum_{i=0}^{a-1}(C_n^i/(i+1)) - \sum_{i=n+b+1}^{n}(C_n^i/(i+1)))$

Переменные a, b идентифицируются с целочисленными константами. Антецеденты обрабатываются проверочными операторами. Уровень срабатывания равен 1.

$\forall_n(\sum_{i=0}^n(C_n^i(-1)^i/(i+1)) = 1/(n+1))$

Уровень срабатывания равен 1.

$\forall_{mn}(\sum_{i=0}^n C_n^i/(i+m) = \int_0^1 t^{m-1}(1+t)^n dt)$

Переменная m идентифицируется с натуральной константой. Интеграл вычисляется с помощью вспомогательной задачи на преобразование. Уровень срабатывания равен 2.

$\forall_{abmn}(0 \leq a \,\&\, b \leq 0 \to \sum_{i=a}^{n+b}(-1)^i C_n^i/(i+m) = \int_0^1 t^{m-1}(1-t)^n dt - \sum_{i=0}^{a-1}(-1)^i C_n^i/(i+m) - \sum_{i=n+b+1}^n(-1)^i C_n^i/(i+m))$

Переменные a, b идентифицируются с целочисленными константами, m - с натуральной константой. Антецеденты выделены указателем "программа". Уровень срабатывания равен 2.

$\forall_{abn}(b \leq 0 \to \sum_{i=a}^{n+b}(-1)^i C_n^i/i = -\sum_{i=1}^n 1/i - \sum_{i=1}^{a-1}(-1)^i C_n^i/i - \sum_{i=n+b+1}^n(-1)^i C_n^i/i)$

Переменная a идентифицируется с натуральной константой, b - с целочисленной константой. Антецедент выделен указателем "программа". Уровень срабатывания равен 2.

7. Сумма частных от деления двух чисел сочетаний.

   $\forall_{mn}(0 < m - n \to \sum_{i=0}^n C_n^i/C_m^i = (m+1)/(m-n+1))$

   Антецедент обрабатывается проверочным оператором. Уровень срабатывания равен 1.

8. Сумма квадратов чисел сочетаний.

   $\forall_n(n - \text{целое} \to \sum_{i=0}^n(C_n^i)^2 = C_{2n}^n)$

   Аналогично предыдущему.

9. Изменение направления суммирования.

   $\forall_{mnpf}(p = m - n \to \sum_{k=n}^m(C_p^{m-k}f(k)) = \sum_{i=0}^{m-n}(C_p^i f(m-i)))$

   Переменная f функциональная. Антецедент выделен указателем "идентификатор". Уровень срабатывания равен 3.

10. Суммирование по четным либо нечетным значениям.

    $\forall_{mn}(m - [n/2] = 0 \to \sum_{i=0}^m C_n^{2i} = 2^{n-1})$

    $\forall_{mn}(m - [(n-1)/2] = 0 \to \sum_{i=0}^m C_n^{2i+1} = 2^{n-1})$

    Антецедент выделен указателем "идентификатор". Его левая часть обрабатывается нормализаторами общей стандартизации. Уровень срабатывания приема равен 1.

    $\forall_{mn}(m - [n/2] = 0 \to \sum_{i=k}^m C_n^{2i} = 2^{n-1} - \sum_{j=0}^{k-1} C_n^{2j})$

    Переменная k идентифицируется с натуральной константой. Конечная сумма в заменяющем терме выписывается как обычная сумма. Антецедент выделен указателем "идентификатор". Уровень срабатывания равен 2.

11. Суммирование по значениям, кратным заданного натурального числа.

    $\forall_{mnpr}(0 \leq r \,\&\, 0 < m - r \,\&\, p - [(n-r)/m] = 0 \rightarrow \sum_{i=0}^{p} C_n^{mi+r} = 1/m \sum_{j=0}^{m-1}(\cos(\pi((j \text{ при } 0 \leq m-2j, \text{ иначе } j-m)n-2rj)/m)(2+2\cos(2\pi j/m)^{n/2})))$

    Переменная r идентифицируется с целочисленной константой, переменная m - с натуральной константой. Первые два антецедента выделены указателем "программа", последний - указателем "идентификатор". Конечная сумма в заменяюще терме выписывается как обычная сумма. Уровень срабатывания приема равен 2.

12. Знакопеременная сумма сочетаний.

    $\forall_{kn}(0 < n - k \rightarrow \sum_{i=0}^{k}((-1)^{n-i} C_n^i) = C_{n-1}^k)$

    Антецедент обрабатывается проверочным оператором. Уровень срабатывания равен 1.

Суммирование условного выражения

$\forall_{fgmnk}(0 \leq n - k \rightarrow \sum_{i=m}^{n}(f(i) \text{ при } i \leq k, \text{ иначе } g(i)) = \sum_{i=m}^{k} f(i) + \sum_{i=k+1}^{n} g(i))$

Переменные f, g функциональные. Антецедент обрабатывается проверочным оператором. Уровень срабатывания равен 1.

$\forall_{abcfgmnp}(a-\text{целое} \,\&\, b-\text{целое} \,\&\, c-\text{целое} \,\&\, \neg(a=0) \,\&\, p = (c-b)/a \,\&\, 0 \leq p-m \,\&\, 0 \leq n - p \rightarrow \sum_{i=m}^{n}(f(i) \text{ при } ai + b = c, \text{ иначе } g(i)) = f(p) + \sum_{i=m}^{p-1} g(i) + \sum_{i=p+1}^{n} g(i))$

Сумма входит в условие задачи на преобразование. Переменные f, g функциональные. Пятый антецедент выделен указателем "идентификатор", остальные - обрабатываются проверочными операторами. Уровень срабатывания равен 2.

$\forall_{abcfgmn}(\neg(a = 0) \,\&\, 0 < a(c - b - an) \rightarrow \sum_{i=m}^{n}(f(i) \text{ при } ai + b = c, \text{ иначе } g(i)) = \sum_{i=m}^{n} g(i))$

$\forall_{abcfgmn}(\neg(a = 0) \,\&\, 0 < a(am + b - c) \rightarrow \sum_{i=m}^{n}(f(i) \text{ при } ai + b = c, \text{ иначе } g(i)) = \sum_{i=m}^{n} g(i))$

Антецеденты обрабатываются проверочными операторами. Переменные f, g функциональные. Уровень срабатывания равен 3.

$\forall_{fgkmn}(0 \leq k - m \,\&\, k - \text{целое} \,\&\, a = \sum_{i=m}^{k-1} g(i) \,\&\, b = \sum_{i=k+1}^{n} g(i) \,\&\, c = \sum_{i=m}^{n} g(i) \rightarrow \sum_{i=m}^{n}(f(i) \text{ при } i = k, \text{ иначе } g(i)) = (f(k) + a + b \text{ при } k \leq n, \text{ иначе } c))$

Сумма входит в условие задачи на преобразование. Переменные f, g функциональные. Первые два антецедента обрабатываются проверочными операторами, остальные - выделены указателем "идентификатор". Их правые части обрабатываются вспомогательными задачами на упрощение, причем результирующие выражения a, b, c не содержат символа "суммавсех". Уровень срабатывания равен 4.

$\forall_{aPQ}(\sum_{i,P(i)}(0 \text{ при } Q(i), \text{ иначе } a(i)) = \sum_{i,P(i),\neg(Q(i))} a(i))$

$\forall_{aPQ}(\sum_{i,P(i)}(a(i) \text{ при } Q(i), \text{ иначе } 0) = \sum_{i,P(i),Q(i)} a(i))$

Переменные a, P, Q функциональные. Переменная i идентифицируется со связывающей приставкой произвольной длины. Уровень срабатывания равен 3.

$\forall_{abcdP}(\neg(a = 0) \rightarrow \sum_{i,P(i),i-\text{целое}}(d(i) \text{ при } ai+b = c, \text{ иначе } 0) = (d((c-b)/a) \text{ при } P((c-b)/a), \text{ иначе } 0))$

Антецедент обрабатывается проверочным оператором. Переменные d, P функциональные. Уровень срабатывания равен 1.

$\forall_{abcdefghmnP}(\neg(a = 0)\ \&\ \neg(d = 0)\ \&\ g = (c - b)/a\ \&\ h = (f - e)/d\ \&\ \neg(g - h = 0) \rightarrow \sum_{i,P(i),i-\text{целое}}(m(i)$ при $ai + b = c,$ иначе $(n(i)$ при $di + e = f,$ иначе $0)) = (m(g)$ при $P(g),$ иначе $0) + (n(h)$ при $P(h),$ иначе $0))$

Третий и четвертый антецеденты выделены указателем "идентификатор", остальные - обрабатываются проверочными операторами. Переменные m, n, P функциональные. Уровень срабатывания равен 1.

$\forall_{PQR}(\sum_{i,P(i),Q} R(i) = (\sum_{i,P(i)} R(i)$ при $Q,$ иначе $0))$

Переменные P, R функциональные. Уровень срабатывания равен 0.

$\forall_{abfmnp}(n-$натуральное $\&\ p = -[-m/2]\ \&\ k = -[-(m+1)/2] \rightarrow \sum_{i=m}^{n} f(i, (a(i)$ при $i-$even, иначе $b(i))) = \sum_{i=p}^{[n/2]} f(2i, a(2i)) + \sum_{i=k}^{n-[n/2]} f(2i - 1, b(2i - 1)))$

Переменные a, b, f функциональные. Указатель "вхождение(f)" означает, что идентификация шаблона $f(i, \ldots)$ происходит путем выделения внутри выражения под суммой некоторого вхождения условного выражения $(a(i)$ при $i -$ even, иначе $b(i))$. Первый антецедент обрабатывается проверочным оператором, остальные - выделены указателем "идентификатор". Уровень срабатывания равен 2.

$\forall_{abfmpqAPQ}(A(m)\ \&\ p = \sum_{i,A(i),\neg(i=m)} f(i, (a(i)$ при $Q(i),$ иначе $b(i)))\ \&\ q = \sum_{i,A(i)} f(i, (a(i)$ при $Q(i),$ иначе $b(i))) \rightarrow \sum_{i,A(i)} f(i, (a(i)$ при $i = m\ \&\ P(i) \vee Q(i),$ иначе $b(i))) = (f(m, a(m)) + p$ при $P(m),$ иначе $q))$

Конечная сумма входит в условие задачи на преобразование. Переменные $a, b, A, P, Q,$ f функциональные. Указатель "вхождение(x6)" означает, что идентификация шаблона $f(i, \ldots)$ происходит при усмотрении внутри общего члена суммы вхождения условного выражения $(a(i)$ при $i = m\ \&\ P(i)\ \vee\ Q(i),$ иначе $b(i))$. При формировании терма $f(X, Y)$ это вхождение заменяется на терм Y, а все расположенные вне него вхождения переменной i заменяются на X. Первый антецедент обрабатывается вспомогательной задачей на доказательство со слабым ограничителем трудоемкости, второй и третий - выделены указателем "идентификатор". Их правые части обрабатываются вспомогательными задачами на преобразование. Проверяется, что результаты p, q не содержат символа "суммавсех". Уровень срабатывания равен 3.

$\forall_{abfmnAB}(m = \sum_{i,A(i),B(i)} f(i, a(i))\ \&\ n = \sum_{i,A(i),\neg B(i)} f(i, b(i)) \rightarrow \sum_{i,A(i)} f(i, (a(i)$ при $B(i),$ иначе $b(i))) = m + n)$

Переменная i должна входить в условие $B(i)$. Переменные a, b, A, B, f функциональные. При разложении в ряды Тейлора и Фурье прием блокируется. Указатель "вхождение(x6)" используется так же, как в предыдущем приеме. Антецеденты выделены указателем "идентификатор". Уровень срабатывания равен 6.

$\forall_{abmnP}(0 \leq n - m \rightarrow \sum_{i=m}^{n} P(i, (a(i)$ при $i = m,$ иначе $b(i))) = P(m, a(m)) + \sum_{i=m+1}^{n} P(i, b(i)))$

$\forall_{abmnP}(0 \leq n - m \rightarrow \sum_{i=m}^{n} P(i, (a(i)$ при $i = n,$ иначе $b(i))) = P(n, a(n)) + \sum_{i=m}^{n-1} P(i, b(i)))$

Конечная сумма входит в условие задачи на преобразование. Переменные a, b, P функциональные. При идентификации шаблона $P(i, \ldots)$ используется указатель "вхождение(x40)". Антецедент обрабатывается проверочным оператором. Уровень срабатывания равен 4. На первой теореме создана еще одна версия приема, применяемая в условиях задач на описание, имеющих цель "вычисление". Такие задачи используются при автоматическом программировании: они позволяют получить список утверждений, преобразуемый компилятором ГЕНОЛОГа в ЛОС-программу. Подробнее эти задачи будут рассмотрены в разделах, связанных с "обычными" вычислениями на ГЕНОЛОГе.

Суммирование по объединению непересекающихся множеств

$\forall_{afAB}(\sum_{x,x\in A} f(x) = a\ \&\ \text{непересек}(A,B) \to \sum_{x,x\in A\cup B} f(x) = a + \sum_{x,x\in B} f(x))$

Переменная f функциональная. Первый антецедент выделен указателем "идентификатор". Его левая часть обрабатывается нормализатором "нормсуммавсех", причем результат a не должен содержать символа "суммавсех". Второй антецедент обрабатывается проверочным оператором. Уровень срабатывания равен 4.

$\forall_{PQR}(\neg Q \to \sum_{i,P\ \vee\ Q,S(i)} R(i) = \sum_{i,P,S(i)} R(i) + \sum_{i,Q,S(i)} R(i))$

Первый антецедент усматривает несовместность условий P и Q. Он обрабатывается вспомогательной задачей на доказательство, которой передается дополнительная посылка P. Введен средний ограничитель трудоемкости. Указатель "содержится(x9 x40 x41)" разрешает переменной i входить в P, Q. Переменные R, S функциональные. Уровень срабатывания равен 2.

Суммы с мощностями множеств

$\forall_{APQ}(\text{конечное}(A) \to \sum_{i,P(i)} Q(\text{card}(A \setminus i)) = \sum_{j=0}^{\text{card}A}(Q(\text{card}A - j)\text{card}(\text{set}_i(P(i)$
$\&\ \text{card}(A \cap i) = j))))$

Прием переходит к суммированию по мощностям пересечений с множеством A множеств i, удовлетворяющих условию $P(i)$. Переменные P, Q функциональные. Указатель "контекст(позиция(x1 фикс(0 1 1 3))вид(x1 мощность(разность(x26 x9))))" определяет усмотрение под суммой выражения $\text{card}(A \setminus i)$. Затем указатель "новаргумент(x41 x9 фикс)" проверяет, что внутри выражения под суммой переменная i встречается только в виде $\text{card}(A \setminus i)$, и таким образом идентифицирует шаблон $Q(\ldots)$. Антецедент обрабатывается проверочным оператором. Уровень срабатывания равен 2.

$\forall_{Afn}(n = \text{card}A \to \sum_{b,b\subseteq A} f(\text{card}\ b) = \sum_{i=0}^{n}(C_n^i f(i)))$

Переменная f функциональная. Указатель "новаргумент(x6 x2 фикс)" проверяет, что в выражении под суммой переменная b встречается только в виде $\text{card}\ b$. Это позволяет идентифицировать шаблон $f(\ldots)$. Антецедент выделен указателем "идентификатор". Его правая часть обрабатывается нормализатором "норммощность". Уровень срабатывания равен 1.

Суммирование по кратным значениям

$\forall_{abkf}(a\text{—натуральное}\ \&\ b\text{—натуральное} \to \sum_{m,k|m,a\leq m,m\leq b} f(m) = \sum_{i=-[-a/k]}^{[} b/k]f(ki))$

Переменная k идентифицируется с натуральной константой. Переменная f функциональная. Антецеденты обрабатываются проверочными операторами. Уровень срабатывания равен 3.

Переход от суммирования по кортежам к суммированию по разбиениям натурального числа

$\forall_{nP}(\sum_{f,\text{кортеж}(f,n,\{1,\ldots,n\})} P(\text{комплект}(f)) =$
$\sum_{s,\text{Разбиение}(n,s)}(P(s)n!C_n^{l(s)}l(s)!)/(\prod_{i=1}^{l(s)} s(i)! \prod_{i,i\in \mathrm{Val}(s)} \mathrm{card}(\text{слой}(s,i))!))$

Преобразуемая сумма берется по всем наборам f длины n, составленным из (возможно, повторяющихся) элементов множества $\{1,\ldots,n\}$. Выражение "комплект(f)" обозначает набор упорядоченных по невозрастанию кратностей элементов набора f. Переменная P функциональная. Указатель "новаругмент(x40 x6 фикс)" обеспечивает проверку того, что f встречается под суммой только в виде "комплект(f)". Заменяющая сумма берется по всем наборам s упорядоченных по невозрастанию натуральных чисел, сумма которых равна n. Уровень срабатывания равен 2.

Двойное суммирование

$\forall_{an}(\sum_{i,j,i<j,i\in\{1,\ldots,n\},j\in\{1,\ldots,n\}} a(i,j) = \sum_{k=1}^{n}\sum_{l=1}^{k-1} a(l,k))$

Прием разворачивает конечную сумму с двумя варьируемыми параметрами в обычную сумму. Переменная n идентифицируется с натуральной константой, меньшей 4. Переменная a функциональная. Указатель "развертка(фикс(0 2)фикс(0 2 1 3))" обеспечивает указанную развертку. Уровень срабатывания равен 3.

$\forall_{mtPQ}(m(j) = \sum_{i,P(i,j)} t(i,j) \to \sum_{i,j,P(i,j),Q(j)} t(i,j) = \sum_{j,Q(j)} m(j))$

Прием выполняет переход от суммирования по двум параметрам к суммированию по одному параметру. Переменные m, t, P, Q функциональные, причем $Q(j)$ идентифицируется с конъюнкцией всех условий на область суммирования, не содержащих i. Антецедент выделен указателем "идентификатор". Его правая часть обрабатывается вспомогательной задачей, выполняющей суммирование по i. Проверяется, что результат $m(j)$ не содержит символа "суммавсех". Указатель "элемент(x9)" разрешает брать в качестве i как первую, так и вторую переменную связывающей приставки. Уровень срабатывания равен 5.

Попытка вычисления суммы в задаче на исследование

$\forall_{afmn}(\sum_{i=m}^{n} f(i) = a \to \sum_{i=m}^{n} f(i) = a)$

Прием имеет заголовок "второйтерм". Конечная сумма входит в посылку задачи на исследование, имеющей цель "известно". Переменная f функциональная, причем общий член суммы не содержит символа "значение". Антецедент выделен указателем "идентификатор". Его левая часть обрабатывается вспомогательной задачей на упрощение. Проверяется, что результат a не содержит символа "суммавсех". Уровень срабатывания равен 3.

Использование индуктивной посылки при доказательстве неравенства

$\forall_{abcf}(0 \le \sum_{n=1}^{c} f(n) + a \;\&\; 0 \le f(c+1) + b - a \to 0 \le \sum_{n=1}^{c+1} f(n) + b)$

Неравенство является условием задачи на доказательство, имеющей комментарий вида (натуральное m). Такие комментарии вводятся во вспомогательных задачах на доказательство шага индукции по параметру m. Первый антецедент идентифицируется с посылкой задачи, истинность второго устанавливается при помощи вспомогательной задачи на доказательство. Указатель "альтернатива" разрешает обработку строгих неравенств. Уровень срабатывания равен 1.

Выделение последнего слагаемого в задачах на доказательство по индукции

$\forall_{afx}(0 < x - a \to \sum_{n=a}^{x} f(n) = f(x) + \sum_{n=a}^{x-1} f(n))$

Сумма входит в условие задачи на доказательство, имеющей комментарий (натуральное m), причем переменная m встречается в выражении x. Выражения $f(x)$ и $\sum_{n=a}^{x-1} f(n)$ до преобразования уже имеются в условии задачи. Переменная f функциональная. Антецедент обрабатывается проверочным оператором. Уровень срабатывания равен 2.

Использование неопределенных сумм

Приемы этого раздела и связанный с ними нормализатор "нормнеопрсумма" реализованы П.А.Пантелеевым. Здесь для суммирования используется подход, аналогичный вычислению определенных интегралов при помощи неопределенных. Находится "дискретная первообразная", и из значения ее в увеличенном на единицу верхнем пределе суммирования вычитается значение в нижнем пределе. Для получения дискретной первообразной служит описываемый в последующих разделах нормализатор "нормнеопрсумма".

1. Обращение к нормализатору "нормнеопрсумма".

   $\forall_{fgmn}(\text{неопрсумма}(\lambda_x(f(x), x - \text{целое})) = \lambda_y(g(y), y - \text{целое})\ \&\ m - \text{целое}\ \&\ n - \text{целое}\ \&\ h(y) = g(y)\ \&\ 0 \leq n - m \to \sum_{i=m}^{n} f(i) = h(n+1) - h(m))$

   Выражение "неопрсумма$(\lambda_x(f(x), x - \text{целое}))$" имеет своим значением некоторую "дискретную первообразную" функции f. Переменные f, g, h функциональные. Первый антецедент выделен указателем "идентификатор". Его левая часть обрабатывается нормализатором "нормнеопрсумма", вычисляющим дискретную первообразную. Четвертый антецедент тоже выделен указателем "идентификатор"; он выполняет дополнительное упрощение найденной первообразной с помощью вспомогательной задачи на преобразование. Остальные антецеденты обрабатываются проверочными операторами. Заменяющий терм упрощается с помощью вспомогательной задачи. Уровень срабатывания приема равен 4.

2. Суммирование произведений экспоненты на синус либо косинус. В этом случае дискретная первообразная выписывается сразу:

   $\forall_{abcdmn}(\neg(a/(\pi b) - \text{целое})\ \&\ m - \text{целое}\ \&\ n - \text{целое}\ \&\ f(k) = c^k(c\sin(a(1-k)/b - d) + \sin(ak/b+d))/(2c\cos(a/b) - c^2 - 1) \to \sum_{k=m}^{n}(c^k \sin(ak/b+d)) = f(n+1) - f(m))$

   $\forall_{abcdmn}(\neg(a/(\pi b) - \text{целое})\ \&\ m - \text{целое}\ \&\ n - \text{целое}\ \&\ f(k) = c^k(c\cos(a(1-k)/b - d) - \cos(ak/b+d))/(c^2 + 1 - 2c\cos(a/b)) \to \sum_{k=m}^{n}(c^k \cos(ak/b+d)) = f(n+1) - f(m))$

   Переменная f функциональная. Первые три антецедента обрабатываются проверочными операторами, четвертый - выделен указателем "идентификатор".

Конечная сумма входит в условие задачи на преобразование. Уровень срабатывания равен 1.

3. Суммирование произведения многочлена на показательную функцию.

   $\forall_{amnP}(\neg(a-1=0)\ \&\ m - \text{целое}\ \&\ n - \text{целое}\ \&\ 0 \leq n-m \rightarrow \sum_{k=m}^{n}(P(k)a^k) = a^{n+1}/(a-1)P(n+1) - a^m(a-1)P(m) - a/(a-1)\sum_{k=m}^{n}((P(k+1)-P(k))a^k))$

   Сумма входит в условие задачи на преобразование. Переменная P функциональная. Фильтр "контекст(видмногочлена(значение(x40 x11)x11 x2))" обеспечивает проверку того, что выражение $P(k)$ после раскрывания скобок преобразуется к многочлену от переменной k. Если a есть минус единица, прием блокируется. Антецеденты обрабатываются проверочными операторами. Уровень срабатывания равен 2.

Нормализатор общей стандартизации "нормсуммавсех"

Нормализатор имеет следующие приемы, большая часть которых аналогична одноименным приемам сканирования задачи:

1. Суммирование константного выражения.
2. Вынесение константного множителя из-под суммы.
3. Вынесение минуса из-под суммы.
4. Заголовок "плюс" выражения под суммой.
5. Суммирование последовательных целых чисел.
6. Суммирование квадратов последовательных целых чисел.
7. Суммирование кубов последовательных целых чисел.
8. Сумма геометрической прогрессии.
9. Переход к обычной сумме.
10. Нормализация тригонометрических аргументов.

    $\forall_{abcdef}(\sum_{i,f(i)}\cos(ai/b+ci/d+e(i)) = \sum_{i,f(i)}\cos((a/b+c/d)i+e(i)))$

    Переменные e, f функциональные.

11. Суммирование характеристик множества экземпляров некоторого объекта.

    $\forall_{abcdef}(\text{экземпляры}(a,b)\ \&\ c \subseteq a\ \&\ f(b)=d \rightarrow \sum_{x,x\in c} f(x) = \text{card}c \cdot d)$

    Прием был введен при рассмотрении текстовых задач. Первый антецедент берется в посылках. Он означает, что a есть множество экземпляров абстрактного объекта b. Последний антецедент тоже берется в посылках. Переменная f выделена указателем "символ", т.е. идентифицируется с некоторым логическим символом, обозначающим характеристику объекта b. Второй антецедент обрабатывается проверочным оператором.

12. Суммирование условного выражения.

13. Использование кванторного тождества.

    $\forall_{abmP}(b \subseteq a \ \& \ \forall_i(i \in a \to P(i) = m) \to \sum_{j,j\in b} P(j) = m\text{card}b)$

    Кванторное тождество берется в посылках. Первый антецедент обрабатывается проверочным оператором.

Нормализатор вычисления неопределенных сумм "нормнеопрсумма"

1. Константа.

    $\forall_a(\text{неопрсумма}(\lambda_x(a, x - \text{целое})) = \lambda_x(ax, x - \text{целое}))$

2. Сумма функций.

    $\forall_{fgpq}(\lambda_x(p(x), x{-}\text{целое}) = \text{неопрсумма}(\lambda_x(f(x), x{-}\text{целое})) \ \& \ \lambda_x(q(x), x{-}\text{целое}) = \text{неопрсумма}(\lambda_x(g(x), x - \text{целое})) \to \text{неопрсумма}(\lambda_x(f(x) + g(x), x - \text{целое})) = \lambda_x(p(x) + q(x), x - \text{целое}))$

    Переменные f, g, p функциональные. Оба антецедента выделены указателем "идентификатор". Они реализуют рекурсивные обращения к нормализатору для вычисления неопределенных сумм первого слагаемого $f(x)$ и остаточной суммы $g(x)$.

3. Вынесение константного множителя.

    $\forall_{abfg}(\lambda_x(g(x), x - \text{целое}) = \text{неопрсумма}(\lambda_x(f(x), x - \text{целое})) \to$
    $\text{неопрсумма}(\lambda_x(af(x)/b, x - \text{целое})) = \lambda_x(ag(x)/b, x - \text{целое}))$

    Переменные f, g функциональные. Либо a, либо b отлично от единицы. Антецедент выделен указателем "идентификатор" и реализует рекурсивное обращение к нормализатору.

4. Показательная функция.

    $\forall_{apqr}(\text{неопрсумма}(\lambda_x(a^{(px/q)+r}, x - \text{целое})) = \lambda_x((a^r x \text{ при } a = 1 \ \vee \ p = 0,$
    $\text{иначе } a^{(px/q)+r}/(a^{p/q} - 1)), x - \text{целое}))$

5. Степенная функция.

    $\text{неопрсумма}(\lambda_x(x, x - \text{целое})) = \lambda_x(x(x - 1)/2, x - \text{целое})$

    $\text{неопрсумма}(\lambda_x(x^2, x - \text{целое})) = \lambda_x(x(2x - 1)(x - 1)/6, x - \text{целое})$

    $\text{неопрсумма}(\lambda_x(x^3, x - \text{целое})) = \lambda_x((x(x - 1))^2/4, x - \text{целое})$

    $\forall_p(0 \le p - 4 \to \text{неопрсумма}(\lambda_x(x^p, x - \text{целое})) =$
    $\lambda_x(1/(p + 1) \sum_{i=0}^{p+1}((p + 1)!\text{числобернулли}(i)x^{p+1-i})/(i!(p + 1 - i)!), x - \text{целое}))$

    В последнем приеме переменная p идентифицируется с целочисленной константой, причем конечная сумма разворачивается в обычную. Уровень срабатывания этого приема равен 2, в то время как уровень срабатывания прочих описанных выше приемов данного нормализатора равен 1.

6. Знакопеременная степенная функция.

    $\forall_{pl}(l - \text{целое} \to \text{неопрсумма}(\lambda_x((-1)^{x+l}x^p, x - \text{целое})) =$
    $\lambda_x((-1)^{x+l-1} \sum_{i=1}^{p+1}(p!(1 - 2^i)\text{числобернулли}(i)x^{p+1-i})/(i!(p + 1 - i)!), x - \text{целое}))$

    Переменная p идентифицируется с натуральной константой. Конечная сумма разворачивается в обычную. Уровень срабатывания равен 1.

7. Синус и косинус.

   $\forall_{abh}(\text{неопрсумма}(\lambda_x(\sin(ax/b+h), x-\text{целое})) = \lambda_x((x\sin h \text{ при } (a/2\pi b - \text{целое}), \text{иначе } -\cos(ax/b+h-a/(2b))/2\sin(a/(2b))), x-\text{целое}))$

   $\forall_{abh}(\text{неопрсумма}(\lambda_x(\cos(ax/b+h), x-\text{целое})) = \lambda_x((x\cos h \text{ при } (a/2\pi b - \text{целое}), \text{иначе } \sin(ax/b+h-a/(2b))/2\sin(a/(2b))), x-\text{целое}))$

8. Произведение синуса либо косинуса на показательную функцию.

   $\forall_{abch}(\text{неопрсумма}(\lambda_x(c^x\sin(ax/b+h), x-\text{целое})) =$
   $\lambda_x((x\sin h \text{ при}(c=1 \,\&\, (a/(\pi b)-\text{even} \;\vee\; c=-1 \,\&\, \neg(a/(\pi b)-\text{even}))),$
   $\text{иначе } c^x(c\sin(a(1-x)/b-h)+\sin(ax/b+h))/(2c\cos(a/b)-c^2-1)), x-\text{целое}))$

   $\forall_{abch}(\text{неопрсумма}(\lambda_x(c^x\cos(ax/b+h), x-\text{целое})) =$
   $lambda_x((x\cos h \text{ при}(c=1 \,\&\, (a/(\pi b)-\text{even} \;\vee\; c=-1 \,\&\, \neg(a/(\pi b)-\text{even}))),$
   $\text{иначе } c^x(c\cos(a(1-x)/b-h)-\cos(ax/b+h))/(c^2+1-2c\cos(a/b))), x-\text{целое}))$

9. Суммирование по частям.

   $\forall_{fgpu}(\lambda_x(g(x), x-\text{целое}) = \text{неопрсумма}(\lambda_x(f(x), x-\text{целое})) \,\&\, \lambda_x(u(x), x-\text{целое}) = \text{неопрсумма}(\lambda_x(g(x+1)(p(x+1)-p(x)), x-\text{целое})) \to \text{неопрсумма}(\lambda_x(p(x)f(x), x-\text{целое})) = \lambda_x(p(x)g(x)-u(x), x-\text{целое}))$

   Переменные f, g, p, u функциональные. Выражение $p(x)$ идентифицируется с некоторым сомножителем рассматриваемого произведения. Антецеденты выделены указателем "идентификатор" и реализуют рекурсивные обращения к нормализатору. Проверяется, что $p(x)$ после раскрывания скобок превращается в многочлен от x. Уровень срабатывания приема равен 4.

14.2 Приемы символа "произведениевсех"

Выражение "произведениевсех(a)" обозначает произведение всех значений вещественнозначной функции a на ее конечной области определения. Формульный редактор рассчитан лишь на те случаи, когда функция a задана через описатель "отображение". Если выражение имеет вид "произведениевсех(отображение(i ($i-$целое & $m \le i$ & $i \le n$) $A(i)$))", то оно прорисовывается формульным редактором в виде:

$$\prod_{i=m}^{n} A(i).$$

В общем случае выражение "произведениевсех(отображение(i $P(i)$ $A(i)$))" прорисовывается так:

$$\prod_{i,P(i)} A(i).$$

Символ "произведениевсех" характеризуется справочниками "тип" и "развертка" (последний используется компилятором ГЕНОЛОГа при наличии указателя "развертка", определяющего идентификацию либо выписывание конечного произведения как обычного).

Общая стандартизация

1. Перемножение константного выражения.

$$\forall_{af}(\prod_{x,f(x)} a = a^{\mathrm{card}(\mathrm{set}_x(f(x)))})$$

Переменная f функциональная, связывающая приставка x может быть любой длины. Уровень срабатывания равен 0.

2. Показательная функция под произведением.

$$\forall_{afg}(\prod_{x,f(x)} a^{g(x)} = a^{\sum_{x,f(x)} g(x)})$$

Произведение входит в условие задачи на преобразование. Переменные f, g функциональные. Связывающая приставка имеет произвольную длину. Уровень срабатывания равен 1.

3. Дробь под произведением.

$$\forall_{fgh}(\prod_{x,h(x)} \frac{f(x)}{g(x)} = \prod_{x,h(x)} f(x)/\prod_{x,h(x)} g(x))$$

Аналогично предыдущему. Уровень срабатывания равен 2.

4. Перемножение произведений.

$$\forall_{fgmn}(\prod_{i=m}^{n}(f(i)g(i)) = \prod_{i=m}^{n} f(i)\prod_{i=m}^{n} g(i))$$

Произведение входит в условие задачи на преобразование. Переменные f, g, h функциональные. Преобразование блокируется, если произведение расположено внутри утверждения о сходимости. Уровень срабатывания равен 1.

5. Вынесение минуса.

$$\forall_{amn}(\prod_{i=m}^{n}(-a(i)) = (-1)^{n-m+1}\prod_{i=m}^{n} a(i))$$

Переменная a функциональная. Уровень срабатывания равен 1.

6. Сокращение конечных произведений.

$$\forall_{abcdefg}(0 < b - c \to (e(\prod_{n=a}^{b} f(n))^d)/(g(\prod_{n=a}^{c} f(n))^d) = e(\prod_{n=c+1}^{b} f(n))^d/g)$$

Дробь входит в условие задачи. Переменная f функциональная. Допускается перестановка числителя и знаменателя. Антецедент обрабатывается проверочным оператором. Уровень срабатывания равен 1.

7. Разбиение на два подпроизведения для последующего сокращения дроби.

$$\forall_{abcdefg}(0 < e - c \to (a\prod_{n=c}^{d} g(n))/(b\prod_{n=e}^{f} g(n)) = (a\prod_{n=e}^{d} g(n)\prod_{n=c}^{e-1} g(n))/(b\prod_{n=e}^{f} g(n)))$$

$$\forall_{abcdefg}(0 < d - f \to (a\prod_{n=c}^{d} g(n))/(b\prod_{n=e}^{f} g(n)) = (a\prod_{n=c}^{f} g(n)\prod_{n=f+1}^{d} g(n))/(b\prod_{n=e}^{f} g(n)))$$

Дробь входит в условие задачи. Переменная f функциональная. В первом случае проверяется, что не очевидно $d < e$, во втором - что не очевидно $f < c$. Указатель "дробь" разрешает перестановку числителя и знаменателя. Антецедент обрабатывается проверочным оператором. Уровень срабатывания равен 1.

8. Вынесение константного множителя из-под произведения.

$$\forall_{afg}(\prod_{x,f(x)} (ag(x)) = a^{\mathrm{card}(\mathrm{set}_x(f(x)))}\prod_{x,f(x)} g(x))$$

Переменные f, g функциональные. Связывающая приставка имеет произвольную длину. Уровень срабатывания равен 1.

9. Перемножение по конечному отрезку целых чисел.

$$\forall_{amn}(\prod_{i,i\in\{m,\ldots,n\}} a(i) = \prod_{i=m}^{n} a(i))$$

Переменная a функциональная. Уровень срабатывания равен 0.

10. Расширение области перемножения.

$$\forall_{amn}(b = a(n+1) \to b\prod_{i=m}^{n} a(i) = \prod_{i=m}^{n+1} a(i))$$

$$\forall_{amn}(b = a(m-1) \to b\prod_{i=m}^{n} a(i) = \prod_{i=m-1}^{n} a(i))$$

Переменная a функциональная. Антецедент выделен указателем "идентификатор". Его правая часть сквозным образом обрабатывается нормализаторами общей стандартизации. Уровень срабатывания равен 4.

11. Переход к обычному произведению.

$$\forall_{fx}(\prod_{y,y=x} f(y) = f(x))$$

$$\forall_{fx}(\prod_{n=x}^{x} f(n) = f(x))$$

$$\forall_f(\prod_{x,\text{ложь}} f(x) = 1)$$

Переменная f функциональная. Уровень срабатывания равен 0.

$$\forall_{afmk}(k - \text{целое} \; \& \; m - \text{целое} \; \& \; a = m - k \; \& \; 0 < a \to \prod_{n=k}^{m} f(n) = f(k) \prod_{n=k+1}^{m} f(n))$$

Прием реализован в двух версиях: одна применяется в условиях задач на преобразование, другая - в посылках задач на исследование. Третий антецедент выделен указателем "идентификатор". Он вычисляет разность a верхнего и нижнего пределов перемножения. Проверяется, что a есть десятичная константа, которая в случае задач на преобразование не превосходит 4, а в случае задач на исследование - не превосходит 3. Прочие антецеденты обрабатываются проверочными операторами. Переменная f функциональная. В случае задач на исследование требуется, чтобы выражение под произведением содержало неизвестные. Уровень срабатывания в обоих случаях равен 1.

$$\forall_{abcf}(0 < b - c \to \prod_{i=a+b}^{a+c} f(i) = 1)$$

Выражения b, c константные. Антецедент обрабатывается проверочным оператором. Уровень срабатывания равен 2.

$$\forall_{abf}(\neg(a \in \{;f\}) \to \prod_{i,i\in\{a;b\}} f(i) = f(a) \prod_{i,i\in\{;b\}} f(i))$$

Переменная f функциональная. Антецедент обрабатывается проверочным оператором. Уровень срабатывания равен 1.

12. Сдвиг области перемножения.

$$\forall_{abfm}(m - \text{целое} \to \prod_{n=a}^{b} f(n) = \prod_{n=a+m}^{b+m} f(n - m))$$

Созданы две версии приема. Одна срабатывает, если внутри $f(n)$ усматривается выражение $m + n$, у которого m- десятичная константа, причем каждое вхождение n внутри $f(n)$ расположено в сумме, имеющей своим слагаемым десятичную константу, не меньшую m. Другая версия срабатывает, если внутри $f(n)$ усматривается сумма $m+n$, причем каждое вхождение n внутри $f(n)$ расположено в сумме, имеющей своим слагаемым выражение m. Переменная f функциональная. Антецедент обрабатывается проверочным оператором. Уровень срабатывания равен 1.

13. Вынесение множителя для значения параметра, заданного отдельным равенством.

$$\forall_{afgh}(h(a) \to \prod_{x,x=a \,\vee\, f(x),h(x)} g(x) = g(a) \prod_{x,f(x),h(x),\neg(x=a)} g(x))$$

$$\forall_{afgh}(\neg h(a) \to \prod_{x,x=a \,\vee\, f(x),h(x)} g(x) = \prod_{x,f(x),h(x)} g(x))$$

Произведение входит в условие задачи на описание и содержит неизвестные. Переменные f, g, h функциональные. Истинность антецедента устанавливается с помощью вспомогательной задачи на доказательство. Уровень срабатывания равен 1.

14. Изменение направления перемножения.

$$\forall_{fkmnP}(\prod_{j=m}^{n} P(f(k-j)) = \prod_{j=k-n}^{k-m} P(f(j)))$$

Произведение входит в условие задачи на преобразование. Переменная f обычная, P - функциональная. Индекс j встречается в общем члене только в подтермах $f(k-j)$. Идентификация начинается с усмотрения некоторого такого подтерма. Уровень срабатывания равен 6.

15. Усмотрение нулевого сомножителя.

$\forall_{fmxtPQ}((f(x,i) = 0 \ \& \ P(x,i)) = (i = m \ \& \ \neg(x = t)) \to \prod_{i,P(x,i)} f(x,i) = (0 \text{ при } \neg(x = t), \text{ иначе } \prod_{i,P(t,i)} f(t,i)))$

Переменные P, f функциональные. Общий член произведения не содержит конечной суммы. Переменная x идентифицируется с некоторым параметром утверждения $P(x,i)$ либо выражения $f(x,i)$. Антецедент выделен указателем "идентификатор". Его левая часть разрешается относительно i с помощью вспомогательной задачи на описание. В результате должно оказаться, что нулевое значение общего члена произведения достигается в единственной точке m, причем только при x, не равном некоторому значеню t. Введен средний ограничитель трудоемкости. Уровень срабатывания равен 4.

$\forall_{fmP}((f(i) = 0 \ \& \ P(i)) = (i = m) \to \prod_{i,P(i)} f(i) = 0)$

Аналогично предыдущему.

16. Стандартизация произведений, выразимых через двойные факториалы.

$$\forall_n(0 \leq n \to \prod_{i=0}^{n}(-i + \frac{1}{2}) = \frac{(-1)^n \prod_{i=0}^{n-1}(i + \frac{1}{2})}{2})$$

$$\forall_n(0 \leq n \to \prod_{i=0}^{n}(i + \frac{1}{2}) = \frac{\prod_{i=0}^{n}(2i+1)}{2^{n+1}})$$

Антецедент обрабатывается проверочным оператором. Уровень срабатывания равен 3.

Перемножение последовательных натуральных чисел

$$\forall_{mnp}((m+p)-\text{натуральное} \ \& \ (n+p)-\text{целое} \ \& \ 0 \leq n-m+1 \to \prod_{k=m}^{n}(k+p) = \frac{(n+p)!}{(m+p-1)!})$$

Антецеденты обрабатываются проверочными операторами. Уровень срабатывания равен 1.

Перемножение последовательных нечетных натуральных чисел

$\forall_{akln}(l - \text{целое} \ \& \ k - \text{целое} \ \& \ n - \text{целое} \ \& \ (n+1) - \text{even} \ \& \ 0 \leq n+2l-1 \ \& \ a = (n+1)/2 \ \& \ 0 \leq k+a-1 \rightarrow \prod_{m=l}^{k}(2m+n) = ((n+2k)!(l+a-1)!)/(2^{k-l}(n+2l-1)!(k+a-1)!))$

Шестой антецедент выделен указателем "идентификатор", остальные - обрабатываются проверочными операторами. Уровень срабатывания равен 1.

Попытка разложить на множители выражение под произведением

$$\forall_{fgh}(g(x) = f(x) \rightarrow \prod_{x,h(x)} f(x) = \prod_{x,h(x)} g(x))$$

Произведение входит в условие задачи на преобразование. Переменные f, g, h функциональные. Общий член произведения имеет заголовок "плюс". Антецедент выделен указателем "идентификатор". Его правая часть обрабатывается нормализатором "видумножение". Проверяется, что результат имеет (с точностью до знака) заголовок "умножение" либо "степень". Уровень срабатывания равен 4.

Перемножение условного выражения

$\forall_{abmnP}(\prod_{i=m}^{n} P(i, (a(i) \text{ при } i = m, \text{ иначе } b(i))) = P(m, a(m)) \prod_{i=m+1}^{n} P(i, b(i)))$

Произведение входит в условие задачи на преобразование. Переменные a, b, P функциональные. Шаблон $P(i, \ldots)$ идентифицируется по какому-либо вхождению условного подвыражения $(a(i) \text{ при } i = m, \text{ иначе } b(i))$. Уровень срабатывания равен 5.

$\forall_{abmnpP}(0 \leq p - m \ \& \ 0 \leq n - p \rightarrow \prod_{i=m}^{n} P(i, (a(i) \text{ при } i \leq p, \text{ иначе } b(i))) = \prod_{i=m}^{p} P(i, a(i)) \prod_{i=p+1}^{n} P(i, b(i)))$
Аналогично предыдущему, но уровень срабатывания равен 3.

Разбор случаев для вынесения множителя

$\forall_{ah}(h(a) \ \vee \ \neg(h(a)))$

Прием имеет заголовок "выводусловия". Идентификация начинается с усмотрения подвыражения $\prod_{x,(x=a \ \vee \ f(x)) \ \& \ h(x)} g(x)$ в условии задачи на описание. Переменные f, g, h функциональные, причем $h(a)$ содержит неизвестные. Выводимая дизъюнкция сопровождается комментарием "разборслучаев". Уровень срабатывания равен 2.

Кванторная посылка, указывающая число сомножителей, равных данному значению

$\forall_{akPQR}(\forall_j(P(j) \rightarrow \text{card}(\text{set}_i(f(i) = j \ \& \ R(i))) = k(j)) \ \& \ \forall_i(R(i) \rightarrow P(f(i))) \rightarrow \prod_{i,R(i)} Q(f(i)) = \prod_{j,P(j)}(Q(j))^{k(j)})$

$\forall_{fmnkPQ}(\forall_j(P(j) \rightarrow \text{card}(\text{set}_i(f(i) = j \ \& \ i \in \{m, \ldots, n\})) = k(j)) \ \& \ \forall_i(i \in \{m, \ldots, n\} \rightarrow P(f(i))) \rightarrow \prod_{i=m}^{n} Q(f(i)) = \prod_{j,P(j)}(Q(j)^{k(j)}))$

Произведение входит в условие задачи. Антецеденты идентифицируются с кванторными импликациями из контекста. Переменная f обычная, переменные k, P, Q, R

функциональные. Указатель "новаргумент(x41 x9 фикс)" определяет идентификацию шаблона $Q(\ldots)$ по всем вхождениям подвыражения $f(i)$. При этом проверяется, что i входит в общий член произведения только в виде $f(i)$. Переменная f не должна входить в $k(i), P(j)$. Уровень срабатывания равен 3.

Перемножение по области значений отображения

$$\forall_{fmnP}(\text{Отображение}(f, \{m, \ldots, n\}, \{k, \ldots, s\}) \to \prod_{i=m}^{n} P(f(i)) = \prod_{i=k}^{s} (P(i))^{\mathrm{card}(\text{слой}(f,i))})$$

Переменная f обычная, P - функциональная. Указатель "новаргумент(x40 x9 фикс)" определяет идентификацию шаблона $P(\ldots)$ путем выделения всех вхождений подвыражения $f(i)$. Уровень срабатывания равен 2.

Суммирование произведений

$$\forall_{np}(n - \text{натуральное} \to \sum_{i=1}^{n} (p(i) \prod_{j=1}^{i-1} (1 - p(j))) = 1 - \prod_{i=1}^{n} (1 - p(i)))$$

Переменная p функциональная. Уровень срабатывания равен 3.

$$\forall_{fmr}(\sum_{a, a \in \prod_{i=1}^{r} m(i)} \prod_{i=1}^{r} f(a(i)) = \prod_{i=1}^{r} \sum_{j, j \in m(i)} f(j))$$

Переменная a обычная, переменные f, m функциональные. Используется указатель "новаргумент(x6 x1 фикс)". Уровень срабатывания равен 2.

Логарифмирование

$$\forall_{afP}(0 < f(i) \to \log_a(\prod_{i, P(i)} f(i)) = \sum_{i, P(i)} \log_a f(i))$$

Антецедент обрабатывается проверочным оператором, причем вводится дополнительная посылка $P(i)$. Переменные f, P функциональные. Длина связывающей приставки i произвольная. Уровень срабатывания равен 2.

Условие равенства произведения нулю

$$\forall_{aP}(\prod_{i, P(i)} a(i) = 0 \leftrightarrow \exists_i (P(i) \ \& \ a(i) = 0))$$

Переменные a, P функциональные. Длина связывающей приставки i произвольная. Уровень срабатывания равен 2.

$$\forall_{aP}(\neg(\prod_{i, P(i)} a(i) = 0) \leftrightarrow \forall_i (P(i) \to \neg(a(i) = 0)))$$

Аналогично предыдущему, но уровень срабатывания равен 1.

Уравнение с конечным произведением в левой части и нулем в правой

$$\forall_{abf}(\prod_{i=a}^{b} f(i) = 0 \leftrightarrow \exists_j(a \leq j \ \&\ j \leq b \ \&\ j - \text{целое} \ \&\ f(j) = 0))$$

Равенство является условием задачи на описание. Переменная f функциональная. Преобразованное условие снабжается комментарием "серия". Уровень срабатывания равен 1.

Усмотрение положительности конечного произведения

$$\forall_{fg}(\forall_x(g(x) \to 0 < f(x)) \to 0 < \prod_{x,g(x)} f(x))$$

Прием имеет заголовок "второйтерм". Неравенство входит в условие задачи на доказательство. Переменные f, g функциональные. Истинность антецедента устанавливается с помощью вспомогательной задачи на доказательство. Предусмотрена возможность применения приема к нестрогим неравенствам. Уровень срабатывания равен 2.

Сравнение конечного произведения со степенью

$$\forall_{abcdf}(b - a + 1 - d = 0 \ \&\ \forall_n(n \in \{a, \ldots, b\} \to 0 \leq f(n) - c) \ \&\ 0 \leq c \to 0 \leq \prod_{n=a}^{b} f(n) - c^d)$$

Прием имеет заголовок "второйтерм". Неравенство входит в условие задачи на доказательство. Переменная f функциональная. Первый антецедент выделен указателем "идентификатор". Его левая часть обрабатывается нормализатором "стандплюс". Истинность второго антецедента устанавливается с помощью вспомогательной задачи на доказательство. Третий антецедент обрабатывается проверочным оператором. Уровень срабатывания равен 1.

Использование индуктивной посылки при доказательстве неравенства

$\forall_{abcde}(0 \leq e\prod_{n=1}^{c} f(n) + a \ \&\ \neg(e = 0) \ \&\ 0 \leq edf(c+1) \ \&\ 0 \leq b - adf(c+1)/e \to 0 \leq d\prod_{n=1}^{c+1} f(n) + b)$

Неравенство является условием задачи на доказательство, имеющей комментарий (натуральное m), т.е. применяемой для рассмотрения шага индукции по некоторому параметру m. Переменная f функциональная. Первый антецедент берется в посылках. Второй обрабатывается проверочным оператором. Истинность третьего антецедента устанавливается с помощью вспомогательной задачи на доказательство, решаемой до уровня 4, истинность четвертого - с помощью вспомогательной задачи на доказательство, решаемой до максимального уровня текущей задачи. Указатель "альтернатива" разрешает обработку строгих неравенств. Уровень срабатывания равен 1.

Нормализатор общей стандартизации "нормпроизведениевсех"

Нормализатор имеет следующие приемы, дублирующие одноименные приемы сканирования задачи:

5. Перемножение константного выражения.

6. Вынесение константного множителя из-под произведения.

7. Перемножение последовательных натуральных чисел.

8. Переход к обычному произведению.

9. Показательная функция под произведением.

14.3 Приемы символа "числосочетаний"

Выражение "числосочетаний(m n)" обозначает число сочетаний из m элементов по n. Формульный редактор прорисовывает его в виде C_m^n. Символ "числосочетаний" характеризуется справочниками "тип", "одз", "типданных".

Вычисление числа сочетаний

$$\forall_{mn}(m - \text{целое} \ \& \ n - \text{целое} \ \& \ 0 \leq n \ \& \ 0 \leq m - n \rightarrow C_m^n = \frac{m!}{n!(m-n)!})$$

Переменные m, n идентифицируются с десятичными константами, причем m должно быть меньше 60, а n - не меньше 5. Последнее ограничение объясняется тем, что для n меньших 5 имеется другой прием, выражающий число сочетаний через конечное произведение и факториал. Антецеденты обрабатываются проверочными операторами. Уровень срабатывания равен 1.

Нормализация второго операнда

$$\forall_{knp}(C_{n+p}^{n+k} = C_{n+p}^{p-k})$$

На этой теореме созданы две версии приема. В обоих случаях выражение n неконстантное, а уровень срабатывания равен 0. Первый прием проверяет, что выражение $p - k$ после обработки нормализаторами общей стандартизации оказывается более коротким, чем $n + k$. При этом требуется, чтобы число сочетаний не было расположено под описателем "отображение" по переменным, входящим в p и не входящим в n. Второй прием, наоборот, предполагает, что число сочетаний расположено под описателем "отображение". Здесь k, p идентифицируются с суммами всех слагаемых, не зависящих от связывающей приставки описателя.

Нулевой второй операнд

$$\forall_a(C_a^0 = 1)$$

Уровень срабатывания равен 0.

Единичный второй операнд

$$\forall_n(0 \leq n \ \& \ n - \text{целое} \rightarrow C_n^1 = n)$$

Антецеденты обрабатываются проверочными операторами. Уровень срабатывания равен 0.

Равные операнды

$$\forall_n (n - \text{целое} \to C_n^m = 1)$$

Антецедент обрабатывается проверочным оператором. Уровень срабатывания приема равен 0.

Выражение через факториалы и конечные произведения

$$\forall_{mn} (m - \text{натуральное} \ \& \ n - \text{натуральное} \to C_n^m = \frac{\prod_{k=0}^{m-1}(n-k)}{m!})$$

Переменная m идентифицируется с десятичной константой, меньшей 5. Если решается задача на преобразование, имеющая цель "числосочетаний", то ее условие не должно содержать чисел сочетаний с неконстантным вторым операндом. Антецеденты обрабатываются проверочными операторами. Уровень срабатывания равен 2.

$$\forall_{mn} (n - \text{натуральное} \ \& \ m - \text{целое} \ \& \ 0 \leq m \ \& \ 0 \leq n - m \to C_n^m = \frac{n!}{m!(n-m)!})$$

Число сочетаний входит в условие задачи на преобразование либо на описание. Это условие должно содержать еще какое-либо вхождение факториала или числа сочетаний, такое, что если хотя бы одно из данных двух вхождений расположено под конечным произведением либо конечной суммой, то и другое расположено там же. В случае задачи на описание дополнительно требуется, чтобы преобразуемый терм содержал неизвестные. Антецеденты обрабатываются проверочными операторами. Уровень срабатывания равен 3. На той же теореме создана еще одна версия приема, срсбатывающая в условиях задач на преобразование, имеющих цель "асимптоценка". Здесь требуется, чтобы число сочетаний располагалось только под мультипликативными операциями. Уровень срабатывания равен 1.

$$\forall_{mn} (n - \text{целое} \ \& \ m - \text{целое} \ \& \ 0 \leq m \ \& \ 0 \leq n \to C_n^m = (0 \text{ при } n < m, \text{ иначе } \frac{n!}{m!(n-m)!}))$$

Число сочетаний входит в условие задачи на описание и содержит неизвестные. m есть константа 2 либо 3. Антецеденты обрабатываются проверочными операторами.

Сумма двух последовательных чисел сочетаний

$$\forall_{mnkpq} (n - k = 1 \ \& \ (0 < p \ \& \ 0 < q \ \& \ r = \min(p,q) \ \vee \ p < 0 \ \& \ q < 0 \ \& \ r = \max(p,q)) \to pC_m^n + qC_m^k = rC_{m+1}^n + (p-r)C_m^n + (q-r)C_m^k)$$

Переменные p, q идентифицируются с десятичными константами. Первый антецедент выделен указателем "идентификатор". Его левая часть обрабатывается нормализаторами общей стандартизации. Второй антецедент выделен указателем "программа". Уровень срабатывания равен 1.

Линейная комбинация двух чисел сочетаний, параметры которых отличаются на единицу

$$\forall_{abmnkp} (n - k = 1 \ \& \ m - p = 1 \to aC_n^m + bC_k^p = \frac{(an + bm)C_n^m}{n})$$

Антецеденты выделены указателем "идентификатор". Коэффициенты a, b не должны содержать символ "числосочетаний". Уровень срабатывания равен 1.

Сокращение для чисел сочетаний

$$\forall_{abmnkp}(n-k=1\ \&\ m-p=1 \to \frac{aC_n^m}{bC_k^p}=\frac{an}{bm})$$

Указатель "дробь(...)" разрешает переставлять числитель и знаменатель. Антецеденты выделены указателем "идентификатор". Уровень срабатывания равен 3.

$$\forall_{abcdemnkp}(\frac{aC_{m+p}^{n+k}}{e}+b=dC_m^n \to \frac{f\cdot(\frac{aC_{m+p}^{n+k}}{e}+b)}{cC_m^n}=\frac{fd}{c})$$

Переменные k, p идентифицируются с константными выражениями, переменные m, n - с неконстантными. Антецедент выделен указателем "идентификатор". Его левая часть обрабатывается нормализатором "умножчислосочетаний", снабжаемым комментарием (сомножитель C_m^n). Этот нормализатор предпринимает попытку преобразовать выражение к виду произведения с множителем C_m^n. Приемы его будут приведены ниже. Уровень срабатывания приема равен 1.

Произведение двух чисел сочетаний

$$\forall_{mnk}(C_n^m C_p^n = C_p^m C_{p-m}^{p-n})$$

Произведение расположено внутри описателя "отображение", связывающая приставка которого пересекается с параметрами выражения n и не пересекается с параметрами выражений m, p. Уровень срабатывания равен 1.

Нормализатор "умножчислосочетаний"

Нормализатор используется в приведенном выше приеме сокращения. Он выносит "за скобку" заданное число сочетаний и имеет всего три приема:

1. Выделение слагаемого, имеющего своим множителем заданное число сочетаний.

   $$\forall_{abcmn}(b=cC_m^n \to \frac{aC_m^n}{d}+b=(\frac{a}{d}+c)C_m^n)$$

   Имеется комментарий (сомножитель C_m^n). Антецедент выделен указателем "идентификатор". Его левая часть рекурсивным образом обрабатывается тем же самым нормализатором. Уровень срабатывания равен 1.

2. Выделение слагаемого, имеющего своим множителем число сочетаний, параметры которого отличаются от заданных на константы.

   $\forall_{abcdmnpqr}(b=cC_m^n\ \&\ r=((m+p)!/m!)\cdot(n!/(n+q)!)\cdot((m-n)!/(m-n+p-q)!) \to (aC_{m+p}^{n+q}/d)+b=(ar/d+c)C_m^n)$

   Антецеденты выделены указателем "идентификатор". Левая часть первого антецедента обрабатывается тем же самым нормализатором, правая часть второго обрабатывается нормализаторами общей стандартизации. Уровень срабатывания равен 2.

$$\forall_{abcdmnpqr}(r=\frac{(m+p)!}{m!}\cdot\frac{n!}{(n+q)!}\cdot\frac{(m-n)!}{(m-n+p-q)!} \to \frac{aC_{m+p}^{n+q}}{d}=\frac{ar}{d}C_m^n)$$

   Аналогично предыдущему, но без рекурсивного обращения.

Нормализатор общей стандартизации "нормчислосочетаний"

Нормализатор имеет следующие приемы,дублирующие приемы сканирования задачи:

1. Вычисление числа сочетаний
2. Нормализация второго операнда
3. Нулевой второй операнд
4. Выражение через конечное произведение и факториал.

14.4 Приемы символа "факториал"

Выражение "факториал(n)" обозначает факториал целого неотрицательного числа n. Формульный редактор прорисовывает его в виде $n!$. Символ "факториал" характеризуется справочниками "тип", "арность", "одз", "типданных".

Значение в нуле

$0! = 1$

Уровень срабатывания равен 0.

Вычисление факториала натурального числа

$\forall_{mn}(m = n! \to n! = m)$

Переменная n идентифицируется с целочисленной константой. Антецедент выделен указателем "идентификатор". Его правая часть обрабатывается нормализатором "нормфакториал", который, собственно, и вычисляет нужное значение. Во избежание появления неоправданно больших десятичных записей требуется, чтобы n не превосходило 9. Если факториал входит в условие задачи на преобразование, имеющей цель "число", то ограничение ослабляется: n должно не превосходить 200. При необходимости ограничение можно еще более ослабить. Прием имеет два уровня срабатывания - 1 и 4, причем на уровне 1 он срабатывает только при n меньшем 20.

Приведение подобных членов с факториалами

$$\forall_{abcde}(c - d = e \ \& \ 0 < e \to ac! + bd! = (a\prod_{n=1}^{e}(d+n) + b)d!)$$

Первый антецедент выделен указателем "идентификатор". Он вычисляет разность выражений под факториалами. Проверяется, что эта разность e - десятичная константа, не превосходящая 4 и что выражения c, d не являются десятичными константами. Второй антецедент обрабатывается проверочным оператором. Дополнительно требуется, чтобы a, b не содержали символа "числосочетаний". Коэффициент при $d!$ в заменяющем выражении обрабатывается нормализатором раскрывания скобок "стандплюс", прочие подвыражения - нормализаторами общей стандартизации. Уровень срабатывания равен 1.

Выравнивание выражений под факториалами в общей сумме

$$\forall_{abcdefg}(c-d=g \ \& \ 0<g \to a(d!)^f+b(c!)^e=a(d!)^f+b(d!)^e(\prod_{n=1}^{g}(d+n))^e)$$

Сумма входит в условие задачи на преобразование. Первый антецедент выделен указателем "идентификатор", второй - обрабатывается проверочным оператором. Разность g представлена десятичной константой, меньшей 5. Выражения a, b не содержат символа "числосочетаний". Уровень срабатывания равен 2.

$$\forall_{abcd}(0<c-d \to ac!+bd!=d!(b+a\prod_{n=1}^{c-d}(d+n)))$$

Сумма является условием задачи на преобразование, имеющей цель "разложитьнамножители". Другие слагаемые отсутствуют. Антецедент обрабатывается проверочным оператором. Уровень срабатывания равен 3.

Сокращение факториалов

$\forall_{abcdefghpqr}(g=f+p \ \& \ h=f+q \ \& \ r=\min(p,q) \ \& \ 0<b-c \to (d\cdot((a+b)!)^g)/(e\cdot((a+c)!)^h)=(d\cdot((a+b)!)^{p-r}(\prod_{n=c+1}^{b}(a+n))^{f+r})/(e\cdot((a+c)!)^{q-r}))$

Переменные b, c идентифицируются с целочисленными константами. Первые два антецедента выделены указателем "идентификатор". Они определяют общую часть f показателей степени g, h. Остаточные слагаемые p, q должны быть целочисленными константами. Последние два антецедента выделены указателем "программа". Допускается перестановка местами числителя и знаменателя (имеющая смысл из-за ограничения $0<b-c$). Конечное произведение обрабатывается нормализатором "нормпроизведениевсех", и при малом числе сомножителей преобразуется в обычное произведение. Если $|b-c|<20$, то уровень срабатывания равен 1, иначе он равен 5.

Выравнивание выражений под факториалами в знаменателях суммируемых дробей

$\forall_{abcdmnk}(n - \text{целое} \ \& \ 0<k-m \to a/(b\cdot(n+k)!)+c/(d\cdot(n+m)!)=a/(b\cdot(n+m)!(\prod_{i=m+1}^{k}(n+i)))+c/(d\cdot(n+m)!))$

Сумма входит в условие задачи на преобразование. Переменные m, k идентифицируются с целочисленными константами. Антецеденты обрабатываются проверочными операторами. Уровень срабатывания равен 2.

Раскрывание скобок под факториалом

$\forall_{abcd}(d=a(b+c) \to (a(b+c))!=d!)$

Выражение a константное. Антецедент выделен указателем "идентификатор". Его правая часть обрабатывается нормализатором раскрывания скобок "стандплюс". Уровень срабатывания равен 1.

Занесение множителя под факториал

$\forall_{mnk}(m - n = 1 \to n!mk = (n+1)!k)$

Произведение входит в условие задачи на преобразование. Блокируется случай, когда оно является знаменателем одной из нескольких суммируемых дробей. Уровень срабатывания равен 0.

Суммирование произведений чисел на их факториалы

$\forall_{mkpq}(p{-}\text{целое} \,\&\, q{-}\text{целое} \,\&\, m{-}\text{целое} \,\&\, 0 \le p{+}m \,\&\, 0 \le q{-}p \to \sum_{i=p}^{q}((i{+}k)\cdot(i{+}m)!) = (q+m+1)! - (p+m)! + (k-m)\sum_{i=p}^{q}(i+m)!)$

Антецеденты обрабатываются проверочными операторами. Уровень срабатывания равен 2.

Переход к числу сочетаний от произведения двух факториалов, находящихся под знаком суммирования

$$\forall_{abmnpq}(\sum_{k=p}^{q}\frac{a(k)}{b(k)(k!)^m((n-k)!)^m} = \sum_{k=p}^{q}\frac{a(k)(C_n^k)^m}{b(k)(n!)^m})$$

Переменные a, b функциональные. Выражения $a(k), b(k)$ не содержат символа "факториал". Уровень срабатывания равен 3.

Усмотрение числа сочетаний

$$\forall abmn(n - \text{натуральное} \;\&\; m - \text{натуральное} \;\&\; 0 \le n - m \to \frac{an!}{bm!(n-m)!} = \frac{aC_n^m}{b})$$

Текущий терм задачи не должен содержать факториалов вне рассматриваемой дроби. Антецеденты обрабатываются проверочными операторами. Уровень срабатывания равен 4.

$$\forall_{abn}(n - \text{натуральное} \to \frac{a(2n)!}{b(n!)^2} = \frac{aC_{2n}^n}{b})$$

Выражения a, b не содержат символа "факториал". При нахождении асимптотических оценок прием блокируется. Уровень срабатывания равен 2.

$$\forall_{abmnrPQ}(r = \sum_{i=a}^{b}\frac{P(i)C_{m-i}^{n-i}}{Q(i)} \to \sum_{i=a}^{b}\frac{P(i)(m-i)!}{Q(i)(n-i)!} = (m-n)!r)$$

Переменные P, Q функциональные. Антецедент выделен указателем "идентификатор". Его правая часть обрабатывается вспомогательной задачей на упрощение. Проверяется, что результат r не содержит символа "суммавсех". Уровень срабатывания равен 3.

Использование индуктивной посылки при доказательстве неравенств

$$\forall_{abcdefg}(g = e{-}f \,\&\, 0 < g \,\&\, 0 < ac \,\&\, 0 < cf!{+}d \,\&\, 0 \le b{-}\frac{ad\prod_{n=1}^{g}(f+n)}{c} \to 0 < ae!{+}b)$$

Прием имеет заголовок "второйтерм". Усматриваемое неравенство входит в условие задачи на доказательство. Четвертый антецедент идентифицируется с посылкой, имеющей комментарий (целое m). Такой комментарий означает, что посылка представляет собой индуктивное предположение при доказательстве индукцией по параметру m. Проверяется, что m входит в f. Первый антецедент выделен указателем "идентификатор". Он упрощает разность выражений подфакториалами. Результат g должен представлять собой десятичную константу, меньшую 5. Второй и третий антецеденты обрабатываются проверочными операторами, истинность последнего антецедента устанавливается с помощью вспомогательной задачи на доказательство. Уровень срабатывания равен 1.

Нормализатор общей стандартизации "нормфакториал"

Нормализатор имеет следующие приемы:

1. Значение факториала в 0.
2. Значение факториала в 1.
3. Вычисление факториала.

   $\forall_n(n! = n \cdot (n-1)!)$

   Переменная n идентифицируется с натуральной константой, отличной от 1 и меньшей 100. Подвыражение $(n-1)!$ обрабатывается рекурсивным образом с помощью того же самого нормализатора.

14.5 Приемы символа "числобернулли"

Выражение "числобернулли(n)" обозначает число Бернулли с индексом n. Формульным редактором оно прорисовывается так же, как текстовым.

Значения чисел Бернулли с малыми номерами

1. числобернулли(0)=1
2. числобернулли(1)=−1/2
3. числобернулли(2)=1/6
4. числобернулли(3)=0
5. числобернулли(4)=−1/30
6. числобернулли(6)=1/42
7. числобернулли(8)=−1/30
8. числобернулли(10)=5/66
9. числобернулли(12)=−691/2730

Уровень срабатывания приемов равен 0.

Число Бернулли с нечетным номером

$\forall_n(0 \leq n - 2 \ \& \ \neg(n - \text{even}) \to \text{числобернулли}(n) = 0)$

Антецеденты обрабатываются проверочными операторами. Уровень срабатывания равен 1.

Рекуррентное соотношение для чисел Бернулли

$$\forall_k(0 \leq k - 10 \to \text{числобернулли}(k) = -\frac{\sum_{p=0}^{k-1}(\text{числобернулли}(p)C_{k+1}^{p})}{k+1})$$

Прием применяется в условии задачи на преобразование. Переменная k идентифицируется с натуральной константой. Конечная сумма разворачивается в обычную, причем числа Бернулли под ней заблаговременно вычисляются нормализатором "нормчислобернулли", а числа сочетаний - нормализатором "нормчислосочетаний". Антецедент обрабатывается проверочным оператором. Уровень срабатывания равен 4.

Нормализатор общей стандартизации "нормчислобернулли"

Нормализатор имеет дубликаты приведенных выше приемов, обеспечивающих определение чисел Бернулли с заданными номерами. Использование буфера предотвращает повторное вычисление значений, полученных незадолго до этого.

14.6 Приемы символа "числобелла"

Выражение "числобелла(n)" обозначает число Белла с индексом n (равно числу разбиений n-элементного конечного множества). Формульным редактором оно прорисовывается так же, как текстовым. Единственный прием для чисел Белла - обращение к нормализатору "нормчислобелла", вычисляющему это число: $\forall_{mn}(\text{числобелла}(n) = m \to \text{числобелла}(n) = m)$. Переменная n идентифицируется с целым числом, антецедент выделен указателем "идентификатор". Уровень срабатывания равен 2. Нормализатор "нормчислобелла" имеет следующие приемы:

1. Значение в нуле. числобелла$(0) = 1$.
2. Значение в единице. числобелла$(1) = 1$. Уровни срабатывания этого и предыдущего приемов равны 1.
3. Рекуррентное соотношение.

   $\forall_n(\text{числобелла}(n) = \sum_{i=0}^{n-1}(C_{n-1}^{i}\text{числобелла}(i)))$

   Переменная n идентифицируется с натуральной константой. Конечная сумма разворачивается в обычную. Числа сочетаний и числа Белла в заменяющем терме предварительно вычисляются с помощью нормализаторов "нормчислосочетаний" и "нормчислобелла". Уровень срабатывания равен 2.

Глава 15

Приемы, связанные с многочленами

Многочлен от n переменных над кольцом K, коэффициенты которого задаются функцией F, определенной на множестве наборов целых неотрицательных степеней переменных, обозначается выражением "многочлен(n K F)". Фактически, вместо F здесь указывается произвольное ее ограничение на конечное подмножество, вне которого функция равна 0. Многочлены рассматриваются как некоторые абстрактные объекты, не отождествляемые с формулами "многочленного типа". Два многочлена считаются равными, если равны их кольца K, числа переменных n и функции F. Таким образом, K, n и F по многочлену восстанавливаются однозначно. Формульный редактор прорисовывает выражение "многочлен(n веществполе F)" в виде $\mu_{xy\ldots}(A(x,y,\ldots))$, где список переменных всегда начинается с буквы x и имеет ровно n элементов. $A(x,y,\ldots)$ - соответствующая сумма одночленов. Для кольца вычетов по модулю k используется запись $\mu_{xy\ldots}(A(x,y,\ldots),\mathrm{mod}(k))$. Указанные способы прорисовки формульным редактором допустимы лишь при задании функции F в виде "таблица($\{(i_1,\ldots,i_n)\to k_1,\ldots,(j_1,\ldots,j_n)\to k_m\}$)", т.е. при непосредственном перечислении коэффициентов. Заметим, что посредством $a\to b$ формульный редактор прорисовывает функцию "См(a,b)", определенную в единственной точке a и принимающую значение b. Если F задана каким-либо другим способом, то формульный редактор прорисовывает многочлен так же, как текстовый.

Перечислим логические символы, относящиеся к многочленам. За исключением особых случаеы, они прорисовываются формульным редактором так же, как текстовым. Утверждение "Многочлен(P)" означает, что P есть многочлен. Символ "Многочлен" является названием одного из базисных типов объектов. Утверждение "веществмногочлен(P)" означает, что P есть многочлен над полем вещественных чисел. Выражение "коэффициенты(P)" обозначает набор коэффициентов многочлена P от одной переменной, упорядоченных по возрастанию степеней. В случае ненулевого многочлена последний элемент этого набора - ненулевой. Выражение "коэфф(A b)" обозначает значение в точке b многочлена, коэффициенты которого образуют упорядоченный по возрастанию степеней набор A. Выражение "стмногочлена(P)" обозначает степень многочлена P. Формульный редактор прорисовывает его в виде $\deg P$. Выражение "Числперем(P)" обозначает число переменных многочлена. Выражения "младшиечлены(P)", "старшийчлен(P)" обозначают, соответственно, многочлены, полученные из P отбрасыванием старшего члена либо сохранением только старшего члена. Выражение "коэффициентмн(P $(i_1,\ldots,i_n)$)" обозначает коэффициент многочлена P при члене, набор показателей степеней которого равен $(i_1,\ldots,i_n)$. При $n=1$ здесь вместо одноэлементного набора (i_1) берется само число i_1. Выражение "старшкоэфф(P)" обозначает старший коэффициент многочлена P. Выражение "значениемн(P $(a_1, \ldots, a_n)$)" обозначает значение многочлена P на наборе $(a_1, \ldots, a_n)$ значений переменных. Формульный редактор прорисовывает его так же, как значение функции: $P(a_1, \ldots, a_n)$. Выражение "кольцомн(P)" обозначает кольцо, над которым рассматривается многочлен P. Заметим, что если тот же самый многочлен можно рассматривать над подкольцом данного кольца, то формально это будут различные многочлены. Выражение "типмн(P Q)" обозначает результат приведения многочлена P над некоторым кольцом K к виду многочлена над кольцом Q, если все коэффициенты многочлена принадлежат носителю кольца Q (а также в некоторых других естественных случаях). Выражения "нольмн", "единицамн" обозначают, соответственно, нулевой и единичный многочлены от одной переменной над полем вещественных чисел. Выражения "нольМн", "единицаМн" обозначают нулевой и единичный многочлены от одной переменной над полем комплексных чисел. Выражение "Нольмн(n K)" обозначает нулевой многочлен от n переменных над кольцом K. Выражение "минусмн(P)" обозначает многочлен, полученный из P изменением знаков коэффициентов. Выражения "плюсмн($P_1 \ldots P_n$)", "умножениемн($P_1, \ldots, P_n$)" обозначают, соответственно, сумму и произведение многочленов $P_1, \ldots, P_n$. Выражение "домножмн(a P)" обозначает многочлен, получающийся из многочлена P над некоторым кольцом умножением на элемент a этого кольца. Выражение "степеньмн(P m)" обозначает результат возведения многочлена P в натуральную степень m. Формульный редактор прорисовывает операции над многочленами "минусмн", "плюсмн", "умножениемн", "дожмножмн", "степеньмн" так же, как аналогичные операции над числами. Утверждение "частноемн(P_1 P_2 Q_1 Q_2)" означает, что Q_1, Q_2 суть неполное частное и остаток от деления многочлена P_1 на многочлен P_2. Выражение "вычетмн(P_1 P_2)" обозначает остаток от деления многочлена P_1 на многочлен P_2. Выражение "НОД(P_1 P_2)" обозначает многочлен, являющийся наибольшим общим делителем многочленов P_1, P_2 и имеющий единичный старший коэффициент. Выражение "остаткимн(P_1 P_2 R)" обозначает последовательность взятых со знаком минус остатков, получаемых при нахождении наибольшего общего делителя многочленов P_1, P_2 по алгоритму Евклида. Выражение "производнаямн(P)" обозначает формальную производную многочлена P. Утверждение "частноерешмн(f g h x y)" означает, что многочлены f, g, h, x, y удовлетворяют соотношению $fx + gy = h$. Выражения "суммавсехмн(S)", "произведениевсехмн(S)" обозначают, соответственно, конечную сумму и конечное произведение многочленов семейства S. Они прорисовываются формульным редактором так же, как обычные конечная сумма и конечное произведение. Выражение "корнимн(P)" обозначает множество корней вещественного многочлена P от одной переменной. Аналогично, выражение "Корнимн(P)" обозначает множество корней комплекснозначного многочлена P от одной переменной.

Несмотря на обилие понятий, количество введенных для работы с многочленами приемов пока сравнительно невелико, так как действия здесь хорошо алгоритмизированы. Описанный выше способ явного задания многочленов через терм вида "таблица(...)", хотя и оправданный с точки зрения логической формализации, с вычислительной точки зрения оказался крайне неудобным. Преобразования здесь оказываются чрезмерно трудоемкими, хотя в простых случаях решатель и справляется с ними относительно быстро. Вместе с тем, созданные на том же ГЕНОЛОГе и имеющие аналогичные теоремы приемов вычислительные "нелогические" пакеты ускоряют действия с многочленами на два порядка и позволяют получать неплохое быстродействие. В этих пакетах многочлены представляются числовыми массивами, для быстрой работы с которыми служат специальные непосредственно реализуемые через интерпретатор ЛОСа процедуры. Программированию на ГЕНОЛОГе традиционных "нелогических" вычислений будет посвещен раздел в одной из последующих книг второго тома.

15.1 Приемы символа "многочлен"

1. Отбрасывание нулевых членов.

   $\forall_{abcd}(\text{многочлен}(a, b, \text{таблица}\{c \to 0; d\}) = \text{многочлен}(a, b, \text{таблица}\{; d\}))$

   Указатель "список(фикс(0 1 3 1 1))" определяет идентификацию подвыражения $c \to 0$ без учета порядка элементов списка. Фильтр "не(заголовок(x4 пустоеслово))" отбрасывает случай, когда список одноэлементный, т.е. многочлен тождественно равен нулю. Уровень срабатывания равен 0.

2. Ввод одноэлементной таблицы.

   $\forall_{abcd}(\text{многочлен}(a, b, c \to d) = \text{многочлен}(a, b, \text{таблица}\{c \to d\}))$

   Прием преобразует "одночленный" многочлен к стандартному заданию через символ "таблица". Уровень срабатывания равен 0.

3. Устранение отрицательных коэффициентов в многочленах над кольцом вычетов.

   $\forall_{abcdk}(\text{многочлен}(a, \text{mod}(k), \text{таблица}\{b \to -c; d\}) = \text{многочлен}(a, \text{mod}(k),$
   $\text{таблица}\{b \to -c(\text{mod } k); d\}))$

   Здесь посредством $\text{mod}(k)$ формульный редактор прорисовывает обозначение "вычеты(k)" кольца вычетов по модулю k. Указатель "список(...)" разрешает выделение элемента списка членов без учета порядка. Нормализатор "нормвычет" вычисляет значение $-c(\text{mod } k)$. Уровень срабатывания равен 0.

Создан также нормализатор общей стандартизации "норммногочлен". Кроме приемов "отбрасывание нулевых членов" и "ввод одноэлементной таблицы", рассмотренных выше, он имеет еще один прием:

$\forall_{abc}(\text{многочлен}(a, b, \text{таблица}\{c \to 0\}) = \text{многочлен}(a, b, \text{таблица}\{0 \to 0\}))$

15.2 Приемы символа "стмногочлена"

Для определения степени многочлена служит следующий прием:

$\forall_{fn}(\deg(f) = n \to \deg(f) = n)$

Антецедент выделен указателем "идентификатор". Его левая часть обрабатывается реализованным на ЛОСе нормализатором "нормстмногочлена". Проверяется, что результат n - десятичная константа. Уровень срабатывания равен 0.

15.3 Приемы символа "минусмн"

1. Изменение знака многочлена.

$\forall_{abnx}(-\text{многочлен}(x, \text{веществполе}, \text{таблица}\{; \lambda_i(a(i) \to b(i), i \in \{1, \ldots, n\})\}) = \text{многочлен}(x, \text{веществполе}, \text{таблица}\{; \lambda_i(a(i) \to -b(i), i \in \{1, \ldots, n\})\}))$

Минус в заменяемом терме обозначает символ "минусмн", в заменяющем - обычный минус для чисел. Указатель "альтернатива(...)" разрешает применение приема также к многочленам над кольцом целых чисел. Указатель "развертка(...)" определяет идентификацию и выписывание таблиц для членов многочлена применительно к конечным перечням. Переменные a, b функциональные. Уровень срабатывания равен 1. Для многочленов над кольцом вычетов введена следующая версия приема:

$\forall_{abknx}(-\text{многочлен}(x, \text{mod}(k), \text{таблица}\{; \lambda_i(a(i) \to b(i), i \in \{1, \ldots, n\})\}) = \text{многочлен}(x, \text{mod}(k), \text{таблица}\{; \lambda_i(a(i) \to -b(i)(\text{mod } k), i \in \{1, \ldots, n\})\}))$

2. Нормализатор общей стандартизации "нормминусмн". Приемы нормализатора дублируют приведенные выше приемы сканирования задачи.

15.4 Приемы символа "плюсмн"

$\forall_{abcd}(\text{многочлен}(a, b, \text{таблица}\{; c\}) + \text{многочлен}(a, b, \text{таблица}\{; d\}) = \text{многочлен}(a, b, \text{сумма}(c; d, b)))$

Выражение "сумма(A,K)" обозначает функцию, полученную сложением функций конечного набора A, принимающих значения в кольце K. Область определения новой функции равна объединению областей определения слагаемых, а значение в точке равно сумме значений определенных в этой точке элементов набора A. Символ "сумма" рассматривается в разделе оглавления базы приемов, посвященном простейшим свойствам функций. Там введен нормализатор "нормсумма", который фактически и выполняет всю работу по сложению многочленов. Он имеет следующие приемы:

1. Сложение значений в точке.

   $\forall_{abcd}(\text{сумма}(a \to b, a \to c; d, \text{веществполе}) = \text{сумма}(\text{префикс}(a \to b + c, d), \text{веществполе}))$

   Указатель "список(...)" определяет идентификаицю элементов набора без учета порядка. Указатель "альтернатива(...)" разрешает применение приема также для кольца целых чисел. В случае кольца вычетов применяется аналогичный прием:

   $\forall_{abcde}(\text{сумма}(a \to b, a \to c; d, \text{mod}(e)) = \text{сумма}(\text{префикс}(a \to (b + c)(\text{mod } e), d), \text{mod}(e)))$

2. Переход к таблице.

   $\forall_{abn}(\forall_i(i \in \{1, \ldots, n\} \to \forall_j(j \in \{i+1, \ldots, n\} \to \neg(a(i) - a(j) = 0))) \to \text{сумма}(\lambda_i(a(i) \to b(i), i \in \{1, \ldots, n\}), c) = \text{таблица}\{; \lambda_i(a(i) \to b(i), i \in \{1, \ldots, n\})\})$

   Антецедент выделен указателем "блокпроверок". Он убеждается в том, что точки $a(i)$ попарно различны. Указатели "развертка" обеспечивают идентификацию и выписывание описателей "отображение" в виде конечных наборов.

Создан также нормализатор общей стандартизации "нормплюсмн". Он имеет прием, дублирующий указанный выше прием сканирования задачи, а также прием для сложения многочленов над кольцом вычетов по модулю, заданному не десятичной константой, а натуральной степенью такой константы. Отличие состоит лишь в том, что перед сложением многочленов константа явно вычисляется. Прием использовался при разложении многочленов на множители по методу Берлекэмпа-Гензеля.

15.5 Приемы символа "умножениемн"

1. Упорядочение сомножителей. Прием основан на теореме "коммутативно(умножениемн)" и упорядочивает многочлены в произведении лексикографически.

2. Умножение двух многочленов.

   $\forall_{abcdmnx}(\text{многочлен}(x, \text{веществполе}, \text{таблица}\{; \lambda_i(a(i) \to b(i), i \in \{1, \ldots, n\})\}) \cdot$
   $\text{многочлен}(x, \text{веществполе}, \text{таблица}\{; \lambda_j(c(j) \to d(j), j \in \{1, \ldots, m\})\}) =$
   $\text{многочлен}(x, \text{веществполе}, \text{сумма}(\lambda_{ij}(a(i) + c(j) \to b(i)d(j), i \in \{1, \ldots, n\} \ \& \ j \in \{1, \ldots, m\}), \text{веществполе})))$

   Переменные a, b, c, d функциональные. Указатели "развертка" определяют идентификацию и выписывание термов "отображение(...)" в виде конечных наборов. Указатели "альтернатива" распространяют прием на случай кольца целых чисел. Запись $a(i) + c(j)$ изображает терм "плюсфунк($a(i)$, $c(j)$)" и обрабатывается нормализатором "нормплюсфунк", вычисляющим поразрядную сумму элементов наборов $a(i), c(j)$. Символ "плюсфунк" для обозначения суммы двух вещественных функций будет рассматриваться в разделе, посвященном математическому анализу. Нормализатор "нормсумма"выполняет приведение подобных членов и переход к формату "таблица(...)". Уровень срабатывания приема равен 1. В случае кольца вычетов введен аналогичный прием:

   $\forall_{abcdmnx}(\text{многочлен}(x, \text{mod}(k), \text{таблица}\{; \lambda_i(a(i) \to b(i), i \in \{1, \ldots, n\})\}) \cdot$
   $\text{многочлен}(x, \text{mod}(k), \text{таблица}\{; \lambda_j(c(j) \to d(j), j \in \{1, \ldots, m\})\}) =$
   $\text{многочлен}(x, \text{mod}(k), \text{сумма}(\lambda_{ij}(a(i) + c(j) \to (b(i)d(j))(\text{mod } k), i \in \{1, \ldots, n\} \ \& \ j \in \{1, \ldots, m\}), \text{mod}(k))))$

3. Нормализатор общей стандартизации "нормумножениемн". Кроме копий приведенных выше приемов умножения многочленов, нормализатор также имеет прием для умножения на единичный многочлен (символ "единицамн") и прием для умножения многочленов над кольцом вычетов, модуль которого задан в виде натуральной степени натурального числа.

4. Нормализатор устранения вложенных умножений "стандумножмн". Нормализатор имеет единственный прием, выполняющий отбрасывание "внутренних скобок" в произведении многочленов.

15.6 Приемы символа "домножмн"

1. Умножение многочлена на численный коэффициент.

   $\forall_{abdnx}(d\text{многочлен}(x, \text{веществполе}, \text{таблица}\{; \lambda_i(a(i) \to b(i), i \in \{1, \ldots, n\})\}) =$
   $\text{многочлен}(x, \text{веществполе}, \text{таблица}\{; \lambda_i(a(i) \to db(i), i \in \{1, \ldots, n\})\}))$

   Переменные a, b функциональные. Указатели "развертка" определяют идентификацию и выписывание термов "отображение(...)" в виде конечных наборов.

Указатель "альтернатива" распространяет прием на случай кольца целых чисел. Уровень срабатывания равен 1. Для кольца вычетов введен аналогичный прием:

$\forall_{abdnx}(d\text{многочлен}(x, \mathrm{mod}(k), \text{таблица}\{; \lambda_i(a(i) \rightarrow b(i), i \in \{1, \ldots, n\})\}) =$
$\text{многочлен}(x, \mathrm{mod}(k), \text{таблица}\{; \lambda_i(a(i) \rightarrow (db(i))(\mathrm{mod}\ k), i \in \{1, \ldots, n\})\}))$

2. Нормализатор общей стандартизации "нормдомножмн". Кроме копий приведенных выше приемов домножения на численный коэффициент, нормализатор имеет таже прием домножения на единицу и прием домножения для кольца вычетов, модуль которого представлен в виде натуральной степени натурального числа.

15.7 Приемы символа "степеньмн"

В рассмотренных задачах пока понадобились лишь простейшие приемы - для возведения многочленов в нулевую и первую степени, причем оба приема были созданы в нормализаторе общей стандартизации "нормстепеньмн". Прием для возведения в нулевую степень имеет следующую теорему:

$\forall_{abcd}(f = \text{многочлен}(b, c, d) \rightarrow f^0 = \text{многочлен}(b, c, \text{таблица}\{0 \rightarrow 1\}))$

Здесь в качестве c допускается символ "веществполе" либо "комплекснполе". При возведении в первую степень теорема приема имеет вид $\forall_a(a^1 = a)$.

15.8 Приемы символа "частноемн"

Для деления с остатком многочленов от одной переменной создан синтезатор "частноемн". Обращение к нему выполняет следующий прием:

$\forall_{abcd}(\text{частноемн}(a, b, c, d) \rightarrow \text{частноемн}(a, b, x, y) \leftrightarrow x = c \ \&\ y = d)$

Антецедент выделен указателем "значения". Выражения a, b имеют заголовки "многочлен". Уровень срабатывания равен 1. Приемы синтезатора "частноемн" таковы:

1. Деление двух многочленов степени 0.

$\forall_{abcd}(\neg(d = 0) \rightarrow \text{частноемн}(\text{многочлен}(a, b, \text{таблица}\{0 \rightarrow c\}), \text{многочлен}(a, b,$
$\text{таблица}\{0 \rightarrow d\}), \text{многочлен}(a, b, \text{таблица}\{0 \rightarrow c/d\}), \text{многочлен}\{a, b,$
$\text{таблица}\{0 \rightarrow 0\})))$

В качестве b допустимы логические символы "веществполе", "Целые". Антецедент обрабатывается проверочным оператором. Для кольца вычетов имеется аналогичный прием:

$\forall_{acdef}(\neg(d = 0) \ \&\ c = (df)(\mathrm{mod}\ e) \rightarrow \text{частноемн}(\text{многочлен}(a, \mathrm{mod}(e),$
$\text{таблица}\{0 \rightarrow c\}), \text{многочлен}(a, \mathrm{mod}(e), \text{таблица}\{0 \rightarrow d\}), \text{многочлен}(a, \mathrm{mod}(e),$
$\text{таблица}\{0 \rightarrow f\}), \text{многочлен}\{a, \mathrm{mod}(e), \text{таблица}\{0 \rightarrow 0\})))$

Переменные c, d идентифицируются с целочисленными константами, переменная e - с натуральной константой. Антецеденты выделены указателем "программа".

2. Степень делимого меньше степени делителя.

   $\forall_{fgmnxPQR}(f = \text{многочлен}(x, R, P)\ \&\ g = \text{многочлен}(x, R, Q)\ \&\ m = deg(f)\ \&\ n = \deg(g)\ \&\ m < n \to \text{частноемн}(f, g, \text{многочлен}(x, R, \text{таблица}\{0 \to 0\}), f))$

   В качестве R допускаются символы "веществполе", "Целое" либо выражение "вычеты(...)". Первые два антецедента, выделенные указателем "идентификатор", проверяют, что f, g заданы термами "многочлен(...)". Следующие два антецедента, тоже выделенные указателем "идентификатор", вычисляют с помощью нормализатора "нормстмногочлена" степени многочленов f, g. Последний антецедент выделен указателем "программа".

3. Степень делимого не меньше степени делителя.

   $\forall_{abfghmnpqrsPQ}(f = \text{многочлен}(x, \text{веществполе}, P)\ \&\ g = \text{многочлен}(x, \text{веществполе}, Q)\ \&\ m = \deg(f)\ \&\ n = \deg(g)\ \&\ n \leq m\ \&\ P = \text{таблица}\{m \to a; r\}\ \&\ Q = \text{таблица}\{n \to b; s\}\ \&\ \neg(b = 0)\ \&\ \text{частноемн}(f - g \cdot \text{многочлен}(x, \text{веществполе}, \text{таблица}\{m - n \to a/b\}), g, p, q)\ \&\ p = \text{многочлен}(x, \text{веществполе}, \text{таблица}\{; h\}) \to \text{частноемн}(f, g, \text{многочлен}(x, \text{веществполе}, \text{таблица}\{m - n \to a/b; h\}), q))$

   Первые два антецедента, выделенные указателем "идентификатор", определяют подтермы P, Q. Следующие два антецедента, тоже выделенные указателем "идентификатор", вычисляют степени многочленов m, n. Пятый антецедент, выделенный указателем "программа", проверяет, что степень делителя не превосходит степени делимого. Шестой и седьмой антецеденты выделяют в термах P, Q старшие члены многочленов и идентифицируют их коэффициенты a, b. Восьмой антецедент обрабатывается проверочным оператором. Девятый антецедент выделен указателем "значения" - он осуществляет рекурсивное обращение к синтезатору "частноемн" после шага деления многочленов "столбиком". Последний антецедент выделен указателем "идентификатор". Он определяет список h членов неполного частного последующих шагов деления для присоединения к нему члена, возникшего на текущем шаге деления. Указатель "альтернатива" позволяет применять прием также в кольце целых чисел. Для случая кольца вычетов создан аналогичный прием:

   $\forall_{abcfghkmnpqrsPQ}(f = \text{многочлен}(x, \text{mod}(k), P)\ \&\ g = \text{многочлен}(x, \text{mod}(k), Q)\ \&\ m = \deg(f)\ \&\ n = \deg(g)\ \&\ n \leq m\ \&\ P = \text{таблица}\{m \to a; r\}\ \&\ Q = \text{таблица}\{n \to b; s\}\ \&\ \neg(b = 0)\ \&\ \text{частноемн}(f - g \cdot \text{многочлен}(x, \text{mod}(k), \text{таблица}\{m - n \to c\}), g, p, q)\ \&\ p = \text{многочлен}(x, \text{mod}(k), \text{таблица}\{; h\})\ \&\ a = (bc)(\text{mod } k) \to \text{частноемн}(f, g, \text{многочлен}(x, \text{mod}(k), \text{таблица}\{m - n \to a/b; h\}), q))$

15.9 Приемы символа "вычетмн"

Для нахождения остатка от деления многочленов от одной переменной создан синтезатор "вычетмн". Обращение к нему выполняет следующий прием:

$\forall_{abc}(c = a(\text{mod } b) \to a(\text{mod } b) = c)$

Здесь "вычетмн" прорисовывается формульным редактором так же, как "вычет". Проверяется, что результат c не имеет заголовка "вычетмн". Уровень срабатывания равен 1. Приемы нормализатора "нормвычетмн" таковы:

1. Оба многочлена имеют степень ноль.

   $$\forall_{abcd}(\neg(d=0) \rightarrow \text{многочлен}(a, \text{веществполе}, \text{таблица}\{0 \rightarrow c\})(\mathrm{mod}\ \text{многочлен}(a, \text{веществполе}, \text{таблица}\{0 \rightarrow d\})) = \text{многочлен}\{a, \text{веществполе}, \text{таблица}\{0 \rightarrow 0\}\})$$

   Указатель "альтернатива" разрешает применять прием также над кольцом целых чисел. Для кольца вычетов имеется аналогичный прием:

   $$\forall_{abcd}(\neg(d=0) \rightarrow \text{многочлен}(a, \mathrm{mod}(e), \text{таблица}\{0 \rightarrow c\})(\mathrm{mod}\ \text{многочлен}(a, \mathrm{mod}(e), \text{таблица}\{0 \rightarrow d\})) = \text{многочлен}\{a, \mathrm{mod}(e), \text{таблица}\{0 \rightarrow 0\}\})$$

2. Степень первого многочлена меньше степени второго.

   $$\forall_{fgmn}(m = \deg(f)\ \&\ n = \deg(g)\ \&\ m < n \rightarrow f(\mathrm{mod}\ g) = f)$$

   Первые два антецедента выделены указателем "идентификатор", последний - указателем "программа".

3. Степень первого многочлена не меньше степени второго.

   $$\forall_{fgPpqr}(f = \text{многочлен}(x, \text{Целые}, P)\ \&\ m = \deg(f)\ \&\ n = \deg(g)\ \&\ n \leq m\ \&\ a = \text{коэффициентмн}(f, m)\ \&\ b = \text{коэффициентмн}(g, n)\ \&\ g = \text{многочлен}(x, \text{Целые}, \text{таблица}\{; \lambda_i(p(i) \rightarrow q(i), i \in \{1, \ldots, r\})\})\ \&\ \neg(b=0) \rightarrow f(\mathrm{mod}\ g) = (f + \text{многочлен}(x, \text{Целые}, \text{таблица}\{; \lambda_i(p(i) + m - n \rightarrow -q(i)/b, i \in \{1, \ldots, r\})\}))(\mathrm{mod}\ g))$$

   Первый антецедент, выделенный указателем "идентификатор", определяет подтерм P. Следующие два антецедента, тоже выделенные этим указателем, вычисляют степени многочленов. Четвертый антецедент выделен указателем "программа". Пятый и шестой антецеденты, выделенные указателем "идентификатор", находят коэффициенты при старших членах. Седьмой антецедент идентифицирует показатели степени $p(i)$ и соответствующие им коэффициенты $q(i)$ для второго многочлена. Последний антецедент обрабатывается проверочным оператором. Заменяющий терм соответствует одному шагу понижения степени при вычислении остатка. Указатель "альтернатива" разрешает применение приема также над полем вещественных чисел. Для кольца вычетов создан аналогичный прием:

   $$\forall_{fgkPpqr}(f = \text{многочлен}(x, \mathrm{mod}(k), P)\ \&\ m = \deg(f)\ \&\ n = \deg(g)\ \&\ n \leq m\ \&\ a = \text{коэффициентмн}(f, m)\ \&\ b = \text{коэффициентмн}(g, n)\ \&\ g = \text{многочлен}(x, \mathrm{mod}(k), \text{таблица}\{; \lambda_i(p(i) \rightarrow q(i), i \in \{1, \ldots, r\})\})\ \&\ \neg(b=0)\ \&\ a = (bc)(\mathrm{mod}\ k) \rightarrow f(\mathrm{mod}\ g) = (f + \text{многочлен}(x, \mathrm{mod}(k), \text{таблица}\{; \lambda_i(p(i) + m - n \rightarrow -q(i)c(\mathrm{mod}\ k), i \in \{1, \ldots, r\})\}))(\mathrm{mod}\ g))$$

15.10 Приемы символа "НОД"

Для вычисления наибольшего общего делителя двух многочленов создан нормализатор общей стандартизации "нормНОД". Обращение к нему выполняется следующим приемом:

$$\forall_{fgh}(h = \text{НОД}(f, g) \rightarrow \text{НОД}(f, g) = h)$$

Нормализатор "нормНОД" имеет следующие приемы:

1. Устранение дробных коэффициентов.

$\forall_{abcdfghmn}(f =$ многочлен$(a, b,$ таблица$\{c \to m/n; d\}) \,\&\, \neg(n = 0) \,\&\, g = nf \to$ НОД$(f, h) =$ НОД$(g, h))$

В качестве b берется один из символов "веществполе", "Целые". Первый антецедент выделен указателем "идентификатор". Он выбирает в многочлене f член степени c, имеющий дробный коэффициент m/n с наибольшим знаменателем n - десятичной константой. Второй антецедент обрабатывается проверочным оператором. Третий антецедент, выделенный указателем "идентификатор", обеспечивает домножение многочлена на n с помощью нормализатора "нормдомножмн".

2. Деление многочленов для понижения степени.

   $\forall_{abcdefgmnq}(m = \deg(f) \,\&\, n = \deg(g) \,\&\, n \le m \,\&\, q = f(\mathrm{mod}\ g) \,\&\, q =$ многочлен$(a, b,$ таблица$\{c \to d; e\}) \,\&\, \neg(d = 0) \to$ НОД$(f, g) =$ НОД$(g, q))$

   Первые два антецедента, выделенные указателями "идентификатор", определяют степени многочленов. Третий антецедент обрабатывается проверочным оператором. Четвертый антецедент, выделенный указателем "идентификатор", определяет остаток q от деления многочлена f на g. Пятый антецедент, тоже выделенный этим указателем, а также последний антецедент, устанавливают наличие в q члена с ненулевым коэффициентом d. Проверяется, что выражение b имеет своим заголовком один из символов "веществполе", "Целые", "вычеты".

3. Многочлены делятся друг на друга.

   $\forall_{abcdefgmnq}(m = \deg(f) \,\&\, n = \deg(g) \,\&\, n \le m \,\&\, q = f(\mathrm{mod}\ g) \,\&\, q =$ многочлен$(a, b,$ таблица$\{c \to 0\}) \,\&\, g =$ многочлен$(a, b,$ таблица$\{n \to r; s\}) \to$ НОД$(f, g) = 1/r \cdot g)$

   Аналогично предыдущему пункту, но в качестве b допускается только один из символов "веществполе", "Целые". Для кольца вычетов создан отдельный прием:

   $\forall_{abcdefgmnq}(m = \deg(f) \,\&\, n = \deg(g) \,\&\, n \le m \,\&\, q = f(\mathrm{mod}\ g) \,\&\, q =$ многочлен$(a, \mathrm{mod}(b),$ таблица$\{c \to 0\}) \,\&\, g =$ многочлен$(a, \mathrm{mod}(b),$ таблица$\{n \to r; s\}) \,\&\, 1 = (rl)(\mathrm{mod}\ b) \to$ НОД$(f, g) = l \cdot g)$

4. Деление целочисленных коэффициентов на общий делитель.

   $\forall_{abfgnx}(f =$ многочлен$(x,$ Целые, таблица$\{; \lambda_i(a(i) \to b(i), i \in \{1, \dots, n\})\}) \,\&\, m =$ нодвсех$\{; \lambda_i(b(i), i \in \{1, \dots, n\})\} \,\&\, 1 < m \to$ НОД$(f, g) =$ НОД(многочлен$(x,$ Целые, таблица$\{; \lambda_i(a(i) \to b(i)/m, i \in \{1, \dots, n\})\}), g))$

   Указатели "развертка" определяют идентификацию и выписывание выражений "отображение" в виде конечных наборов. Первый антецедент выделен указателем "идентификатор". Он определяет наборы a, b показателей степени и соответствующих коэффициентов многочлена f. Второй антецедент, также выделенный указателем "идентификатор", вычисляет с помощью нормализатора "нормнодвсех" наибольший общий делитель коэффициентов $b(i)$. Третий антецедент обрабатывается проверочным оператором.

15.11 Приемы символа "частноерешмн"

Для решения многочленного линейного уравнения $fx + gy = h$ с известными f, g, h и неизвестными x, y создан следующий прием, обращающийся к синтезатору "частноерешмн":

$\forall_{abdfghpqrxy}$(частноерешмн(f, g, h, a, b) & НОД$(f, g) = p$ & частноемн(f, p, q, m) & частноемн$(g, p, r, n) \rightarrow fx + gy = h \leftrightarrow \exists_t(x = a + tr$ & $y = b - tq))$

Уравнение является условием задачи на описание. Выражения f, g, h не содержат неизвестных, переменные x, y суть неизвестные. Первый антецедент, выделенный указателем "значения", находит частное решение a, b. Второй антецедент, выделенный указателем "идентификатор", вычисляет наибольший общий делитель многочленов f, g. Третий и четвертый антецеденты, выделеннные указателем "значения", вычисляют частные q, r от деления многочленов f, g на их наибольший общий делитель. Преобразованное условие сопровождается комментарием "серия". Уровень срабатывания равен 3.

Синтезатор "частноерешмн" имеет следующие приемы:

1. Понижение степени многочленов.

   $\forall_{fghpqrs}(\deg(f) \le \deg(g)$ & частноемн(g, f, p, q) & $\neg$(старшкоэфф$(q) = 0)$ & частноерешмн$(f, q, h, r, s) \rightarrow$ частноерешмн$(f, g, h, r - ps, s))$

   $\forall_{fghpqrs}(\deg(f) < \deg(g)$ & частноемн(g, f, p, q) & $\neg$(старшкоэфф$(q) = 0)$ & частноерешмн$(q, f, h, s, r) \rightarrow$ частноерешмн$(g, f, h, s, r - ps))$

   Первый и третий антецеденты обрабатываются проверочными операторами. Второй антецедент, выделенный указателем "значения", выполняет деление многочленов с остатком. Четвертый антецедент, тоже выделенный этим указателем, выполняет рекурсивное обращение к синтезатору.

2. Один многочлен делится на другой.

   $\forall_{bfghpqrsx}(\deg(f) \le \deg(g)$ & частноемн(g, f, p, q) & старшкоэфф$(q) = 0$ & частноемн(h, f, r, s) & старшкоэфф$(s) = 0$ & $f =$ многочлен$(x, b, c) \rightarrow$ частноерешмн$(f, g, h, r,$ многочлен$(x, b,$ таблица$\{0 \rightarrow 0\})))$

   $\forall_{bfghpqrsx}(\deg(f) < \deg(g)$ & частноемн(g, f, p, q) & старшкоэфф$(q) = 0$ & частноемн(h, f, r, s) & старшкоэфф$(s) = 0$ & $f =$ многочлен$(x, b, c) \rightarrow$ частноерешмн$(g, f, h,$ многочлен$(x, b,$ таблица$\{0 \rightarrow 0\}), r))$

   Первый антецедент обрабатывается проверочным оператором, второй и четвертый - выделены указателем "значения", остальные - выделены указателем "идентификатор".

15.12 Приемы символа "производнаямн"

Для непосредственного вычисления производной многочлена созданы следующие приемы:

$\forall_{abcdn}$(производнаямн(многочлен$(a,$ веществполе, таблица$\{; \lambda_i(c(i) \rightarrow d(i), i \in \{1, \ldots, n\})\})) =$ многочлен$(a,$ веществполе, таблица$\{; \lambda_i(c(i) - 1 \rightarrow c(i)d(i), i \in \{1, \ldots, n\})\}))$

$\forall_{abcdn}$(производнаямн(многочлен$(a, \mathrm{mod}(m)$, таблица$\{;\lambda_i(c(i) \to d(i), i \in \{1, \ldots, n\})\}$ $)) =$ многочлен$(a, \mathrm{mod}(m)$, таблица$\{;\lambda_i(c(i) - 1 \to (c(i)d(i))(\mathrm{mod}\ m), i \in \{1, \ldots, n\})\}))$

В первом приеме указатель "альтернатива" разрешает также рассмотрение кольца целых чисел. В обоих приемах указатель "развертка" определяет идентификацию и выписывание термов "отображение" в виде конечных наборов. Уровень срабатывания равен 1.

Эти приемы продублированы в нормализаторе общей стандартизации "нормпроизводнаямн".

15.13 Приемы символа "значениемн"

Для вычисления значения многочлена от одной переменной над кольцом вычетов, заданного перечислением коэффициентов, служит следующий прием:

$\forall_{abcn}$(многочлен$(1, \mathrm{mod}(m)$, таблица$\{;\lambda_i(a(i) \to b(i), i \in \{1, \ldots, n\})\})(c) =$
$\sum_{i=1}^{n}(b(i)c^{a(i)})(\mathrm{mod}\ m))$

Переменные a, b функциональные; m, n - натуральные константы. Указатель "развертка" определяет идентификацию терма "отображение" в виде конечного набора и выписывание конечной суммы как обычной. Уровень срабатывания равен 1. Для случая нескольких переменных введен аналогичный прием:

$\forall_{abcmnp}$(многочлен$(p, \mathrm{mod}(m)$, таблица$\{;\lambda_i(a(i) \to b(i), i \in \{1, \ldots, n\})\})(c) =$
$\sum_{i=1}^{n}(b(i)\prod_{j=1}^{p}(c(j))^{(a(i))(j)})(\mathrm{mod}\ m))$

Здесь добавляется конечное произведение, которое также разворачивается в обычное. Особо оговаривается, что p отлично от единицы.

Нормализатор общей стандартизации "нормзначениемн" имеет единственный прием, обеспечивающий вычисление значения многочлена от одной переменной, заданного перечислением коэффициентов:

$\forall_{abcn}$(многочлен(1, веществполе, таблица$\{;\lambda_i(a(i) \to b(i), i \in \{1, \ldots, n\})\})(c) =$
$\sum_{i=1}^{n}(b(i)c^{a(i)}))$

Допускается случай кольца целых чисел.

15.14 Приемы символа "старшкоэфф"

Создан нормализатор общей стандартизации "нормстаршкоэфф", имеющий единственный прием:

$\forall_{fn}(n = \deg(f) \to$ старшкоэфф$(f) =$ коэффициентмн$(f, n))$

Антецедент выделен указателем "идентификатор". Его правая часть обрабатывается нормализатором "нормстмногочлена". Заменяющий терм обрабатывается нормализатором "нормкоэффициентмн".

15.15 Приемы символа "Числперем"

Имеется единственный прием:

$\forall_{abc}$(Числперем(многочлен$(a, b, c)) = a)$

Уровень срабатывания равен 0.

15.16 Приемы символа "коэффициентмн"

Создан нормализатор общей стандартизации "нормкоэффициентмн", используемый для работы с многочленами от одной переменной, заданными стандартным (через терм вида "таблица") перечислением своих членов. Он имеет следующие два приема:

1. Определение коэффициента при члене заданной степени.

   $\forall_{abfnpx}(f = \text{многочлен}(x, p, \text{таблица}\{n \to a; b\}) \to \text{коэффициентмн}(f, n) = a)$

   Антецедент выделен указателем "идентификатор". Указатель "список" определяет выбор нужного члена без учета порядка элементов списка. Уровень срабатывания равен 1.

2. Остаточное определение нулевого коэффициента.

   $\forall_{pfn}(\text{коэффициентмн}(\text{многочлен}(1, p, f), n) = 0)$

   Заголовком выражения p служит один из символов "веществполе", "Целые", "вычеты". Переменная n идентифицируется с десятичной константой. Выражение f имеет заголовок "таблица". Уровень срабатывания приема равен 2.

15.17 Приемы символа "остаткимн"

Для нахождения последовательности взятых со знаком минус остатков, получаемых при нахождении наибольшего общего делителя многочленов f, g по алгоритму Евклида, создан синтезатор "остаткимн". Он имеет следующие приемы:

1. Степень остатка деления многочленов больше нуля.

   $\forall_{fghp}(f(\text{mod } g) = h \;\&\; \neg(\text{старшкоэфф}(h) = 0) \;\&\; \text{остаткимн}(g, -h, p) \to$
   $\text{остаткимн}(f, g, \text{префикс}(f, p)))$

   Первый антецедент выделен указателем "идентификатор". Его левая часть обрабатывается нормализатором "нормвычетмн". Второй антецедент обрабатывается проверочным оператором. Третий антецедент, выделенный указателем "значения", реализует рекурсивное обращение.

2. Многочлены делятся друг на друга.

   $\forall_{fgh}(f(\text{mod } g) = h \;\&\; \text{старшкоэфф}(h) = 0 \to \text{остаткимн}(f, g, (f, g)))$

   Антецеденты выделены указателем "идентификатор".

Синтезатор "остаткимн" мог бы быть использован при вычислении системы Штурма. Однако, фактически в решателе используется его двойник, ориентированный на работу с "нелогическим" форматом задания многочленов. Этот двойник реализован на ГЕНОЛОГе и имеет те же самые теоремы приемов. Подробнее о нем будет написано в одной из последующих книг второго тома.

15.18 Приемы символа "кольцомн"

Для непосредственного определения кольца многочлена служит следующий прием:

$\forall_{abc}(\text{кольцомн}(\text{многочлен}(a, b, c)) = b)$

Уровень срабатывания равен 0. Если многочлен задан с помощью операций над другими многочленами, то в случае колец вычетов используется синтезатор "усммодульмн", определяющий модуль кольца. Обращение к нему реализуется следующим приемом:

$\forall_{ab}(\text{кольцомн}(a) = \text{mod}(b) \to \text{кольцомн}(a) = \text{mod}(b))$

Антецедент выделен указателем "значения". Уровень срабатывания равен 1. Приемы синтезатора "усммодульмн" таковы:

1. Усмотрение из посылок.

   $\forall_{ab}(\text{кольцомн}(a) = \text{mod}(b) \to \text{кольцомн}(a) = \text{mod}(b))$

   Антецедент берется в посылках.

2. Деление многочленов.

   $\forall_{abcde}(\text{частноемн}(a, b, c, d)\ \&\ \text{кольцомн}(a) = \text{mod}(e) \to \text{кольцомн}(c) = \text{mod}(e))$

   $\forall_{abcde}(\text{частноемн}(a, b, c, d)\ \&\ \text{кольцомн}(a) = \text{mod}(e) \to \text{кольцомн}(d) = \text{mod}(e))$

   Первый антецедент берется в посылках, второй - выделен указателем "значения" и выполняет рекурсивное обращение к синтезатору "усммодульмн".

3. Произведение многочленов.

   $\forall_{abc}(\text{кольцомн}(a) = \text{mod}(b) \to \text{кольцомн}(ac) = \text{mod}(b))$

   Запись ac обозначает здесь выражение "умножениемн(a c)". Антецедент выделен указателем "значения" и выполняет рекурсивное обращение.

15.19 Приемы символа "типмн"

Для преобразования многочлена над одним кольцом к виду многочлена над другим кольцом (в осмысленных ситуациях) создан нормализатор общей стандартизации "нормтипмн". Пока он имеет всего четыре приема, которые понадобились для реализации алгоритма разложения многочленов на множители по методу Берлекэмпа-Гензеля:

1. Переход от многочлена над кольцом вычетов к многочлену над целыми числами.

   $\forall_{abp}(\text{типмн}(\text{многочлен}(a, \text{mod}(p), b), \text{Целые}) = \text{многочлен}(a, \text{Целые}, b))$

2. Переход от многочлена над целыми числами к многочлену над кольцом вычетов.

   $\forall_{abcp}(\text{типмн}(\text{многочлен}(a, f, \text{таблица}\{; \lambda_i(b(i) \to c(i), i \in \{1, \ldots, n\})\}), \text{mod}(p)) = \text{многочлен}(a, \text{mod}(p), \text{таблица}\{; \lambda_i(b(i) \to c(i)(\text{mod } p), i \in \{1, \ldots, n\})\}))$

   $\forall_{abcpq}(q = p^m \to \text{типмн}(\text{многочлен}(a, f, \text{таблица}\{; \lambda_i(b(i) \to c(i), i \in \{1, \ldots, n\})\}), \text{mod}(p^m)) = \text{многочлен}(a, \text{mod}(p^m), \text{таблица}\{; \lambda_i(b(i) \to c(i)(\text{mod } q), i \in \{1, \ldots, n\})\}))$

В первом случае модуль кольца вычетов задается явным образом, во втором - в виде степенного выражения.

3. Переход от многочлена над целыми числами к многочлену над вещественными числами.

   $\forall_{abp}(\text{типмн}(\text{многочлен}(a, \text{Целые}, b), \text{веществполе}) = \text{многочлен}(a, \text{веществполе}, b))$

В заключение раздела заметим, что для вычисления вещественных и комплексных корней многочленов, а также разложения многочленов с целыми коэффициентами на множители на ГЕНОЛОГе реализованы процедуры, использующие традиционные "нелогические" структуры данных. Они будут рассмотрены в третьей книге второго тома.

Список литературы

1. А.Черч. Введение в математическую логику. Том 1. М.: ИЛ, 1960, с.484.
2. С.К.Клини. Введение в метаматематику. М.: ИЛ, 1957. 526с.
3. С.К.Клини. Математическая логика. М.: Мир, 1973, 480с.
4. Ч.Чень, Р.Ли. Математическая логика и автоматическое доказательство теорем. М.: Мир, 1983, 360 с.
5. Э.Мендельсон. Введение в математическую логику. М.: Наука, 1971, с.320.
6. Г.Метакидес, А Нероуд. Принципы логики и логического программирования. М.: Факториал, 1998, 288с.
7. И.Братко. Программирование на языке Пролог для искусственного интеллекта. М.: Мир, 1990, 560 с.
8. Дж.Малпас. Реляционный язык Пролог и его применение. М.: Наука, 1990, 464 с.
9. Э.Хант. Искусственный интеллект. М.: Мир, 1978, 558с.
10. Ж.-Л. Лорьер. Системы искусственного интеллекта. М.: Мир, 1991, 568с.
11. E.A.Bender. Mathematical methods in artificial intelligence. Los Alamitos, IEEE, Comp.Society Press, 1996, 638p.
12. Д. Пойа. Математика и правдоподобные рассуждения. М.: Наука, 1975, 463с.
13. Д. Пойа. Математическое открытие. М.: Наука, 1976, 448с.
14. Лавров И.А., Максимова Л.Л., Задачи по теории множеств, математической логике и теории алгоритмов, М., "Наука", 1975, 232 с.
15. Антонов Н.П., Выгодский М.Я., Никитин В.В., Санкин А.И.. Сборник задач по элементарной математике, М., "Наука", 1964, 528 с.
16. Вавилов В.В., Мельников И.И., Олехник С.Н., Пасиченко П.И. Задачи по математике. Алгебра. М., "Наука", 1988, 431 с.

17. Вавилов В.В., Мельников И.И., Олехник С.Н., Пасиченко П.И. Задачи по математике. Уравнения и неравенства. М., "Наука", 1988, 237 с.

18. Потапов М.К., Олехник С.Н., Нестеренко Ю.В. Конкурсные задачи по математике. М., "Наука", 1992, 478 с.

19. Сканави М.И. Сборник задач по математике. М., "Высшая школа", 1988, 431 с.

20. Ваховский Е.Б., Рывкин А.А. Задачи по элементарной математике. М., "Наука", 1971, 360 с.

21. Лидский В.Б., Овсянников Л.В., Тулайков А.Н., Шабунин М.И., Федосов Б.В. Задачи по элементарной математике. М., 1973, 415 с.

22. Сергеев И.Н. Математика. Задачи с ответами и решениями. М., "Высшая школа", 2003, 336 с.

23. Шарыгин И.Ф. Геометрия, 9-11 классы. М., "Дрофа", 1997, 396 с.

24. Шарыгин И.Ф., Гордин Р.К. Сборник задач по геометрии. М., Астрель, 2001, 396 с.

25. Демидович Б.П. Сборник задач и упражнений по математическому анализу. М., "Наука", 1969, 544 с.

26. Виноградова И.А., Олехник С.Н., Садовничий В.А. Задачи и упражнения по математическому анализу. т.1. М., "Высшая школа", 2000, 722 с.

27. Моденов П.С., Пархоменко А.С. Сборник задач по аналитической геометрии. М., "Наука", 1976, 384 с.

28. Филиппов А.Ф. Сборник задач по дифференциальным уравнениям. М., "Наука", 1965, 100 с.

29. Вентцель Е.С., Овчаров Л.А. Задачи и упражнения по теории вероятностей. М., "Высшая школа", 2000, 364 с.

30. Пантелеев А.В., Якимова А.С. Теория функций комплексного переменного и операционное исчисление в примерах и задачах. М., "Высшая школа", 2001, 445 с.

31. Черноуцан А.И. Физика. Задачи с ответами и решениями. М., Книжный дом "Университет", 2001, 335.

32. Подколзин А.С. Об организации баз знаний, ориентированных на автоматическое решение задач. "Дискретная математика", 1990, т.2., вып.1., с. 13-30.

33. Подколзин А.С. Система автоматического решения задач по элементарной алгебре. "Дискретная математика", 1994, т.6., вып.4., с. 35-57.

34. Подколзин А.С. Компьютерный решатель математических задач. ДАН РФ, 1994, т.335, № 4.

35. Подколзин А.С. Компьютерное моделирование процессов решения математических задач. Изд-во ЦПИ при мех.-мат. факультете МГУ, 2001. 235 с.

36. Подколзин А.С. Компьютерное моделирование логических процессов. Том 1. Архитектура и языки решателя задач. М., "Физматлит", 2008. 1022 с.