Лекция 24.

Линейный по памяти непереборный алгоритм
решения двумерной задачи интервального поиска.

1 Введение

Рассматривается двумерная задача интервального поиска, которая состоит в поиске в конечном подмножестве вещественной плоскости всех тех точек, которые попадают в прямоугольник-запрос. В работе [1] приводится алгоритм решения этой задачи, зависящий от параметра, при вариации которого объем памяти, необходимый алгоритму, изменяется от $O(k^3)$ до $O(k \log k)$, при этом среднее время поиска (без учета времени на перечисление ответа) изменяется от $O(1)$ до $O(\log k)$. Здесь k — мощность множества, в котором производится поиск. Как видно в лучшем (по памяти) случае требуется $O(\log k)$ памяти. С другой стороны легко привести пример линейного по памяти алгоритма. Это переборный алгоритм, который по очереди просматривает точки из исходного множества и проверяет на попадание в прямоугольник-запрос. Но среднее время у этого алгоритма равно $O(k)$. В данной работе предлагается линейный по памяти алгоритм, который имеет среднее время поиска (помимо перечисления ответа), равное $O(\sqrt{k})$.

Излагаемые здесь результаты были опубликованы в [2].

2 Основные понятия и формулировка результата

Мы будем использовать терминологию и обозначения из работ [1, 3].

Пусть X — множество запросов с заданным на нем вероятностным пространством $\langle X, \sigma, \mathbf{P} \rangle$, где σ — алгебра подмножеств множества X, $\mathbf{P}$ — вероятностная мера на σ; Y — множество записей (объектов поиска); ρ — бинарное отношение на $X \times Y$, называемое отношением поиска. Пятерку $S = \langle X, Y, \rho, \sigma, \mathbf{P} \rangle$ будем называть *типом*. Тройку

$I = \langle X, V, \rho \rangle$, где V — некоторое конечное подмножество множества Y, в дальнейшем называемое *библиотекой*, будем называть задачей информационного поиска (ЗИП) типа S, и будем считать, что ЗИП $I = \langle X, V, \rho \rangle$ содержательно состоит в перечислении для произвольно взятого запроса $x \in X$ всех тех и только тех записей $y \in V$ таких, что $x \rho y$.

Пусть f — одноместный предикат, определенный на X , то есть $f : X \to \{0, 1\}$. Множество $N_f = \{x \in X : f(x) = 1\}$ назовем *характеристическим множеством* предиката f.

Множество $O(y, \rho) = \{x \in X : x \rho y\}$ назовем *тенью* записи $y \in Y$.

Функцию $\chi_{y,\rho} : X \to \{0, 1\}$ такую, что $N_{\chi_{y,\rho}} = O(y, \rho)$ назовем *характеристической функцией* записи y.

Пусть F — множество символов одноместных предикатов, определенных на множестве X, G — множество символов одноместных переключателей, определенных на множестве X. Под переключателем будем понимать функцию, областью значений которой является конечное подмножество натурального ряда. Пару $\mathcal{F} = \langle F, G \rangle$ назовем базовым множеством.

Понятие *информационного графа* (ИГ) над базовым множеством $\mathcal{F} = \langle F, G \rangle$ определяется следующим образом. Берется конечная многополюсная ориентированная сеть. В ней выбирается некоторый полюс, который называется корнем. Остальные полюса называются листьями и им приписываются записи из Y, причем разным листьям могут быть приписаны одинаковые записи. Некоторые вершины сети (в том числе это могут быть и полюса) называются переключательными и им приписываются переключатели из G. Ребра, исходящие из каждой из переключательных вершин, нумеруются подряд, начиная с 1, и называются переключательными ребрами. Ребра, не являющиеся переключательными, называются предикатными и им приписываются предикаты из множества F. Таким образом нагруженную многополюсную ориентированную сеть называем ИГ над базовым множеством $\mathcal{F} = \langle F, G \rangle$.

Функционирование ИГ определяется следующим образом. Скажем, что предикатное ребро проводит запрос $x \in X$, если предикат, приписанный этому ребру, принимает значение 1 на запросе x, Переключательное ребро, которому приписан номер n, проводит запрос $x \in X$, если переключатель, приписанный началу этого ребра, принимает значение n на запросе x. Ориентированная цепочка ребер проводит запрос $x \in X$, если каждое ребро цепочки проводит запрос x. Запрос $x \in$

X проходит в вершину β ИГ, если существует ориентированная цепочка, ведущая из корня в вершину β, которая проводит запрос x. Запись y, приписанная листу α, попадает в ответ ИГ на запрос $x \in X$, если запрос x проходит в лист α. Ответом ИГ U на запрос x назовем множество записей, попавших в ответ ИГ на запрос x, и обозначим его $\mathcal{J}_U(x)$. Эту функцию $\mathcal{J}_U(x)$ будем считать результатом функционирования ИГ U.

Пусть нам дана ЗИП $I = \langle X, V, \rho \rangle$. Скажем, что ИГ U *решает* ЗИП $I = \langle X, V, \rho \rangle$, если $\mathcal{J}_U(x) = \{y \in V : x\rho y\}$.

Введем понятие *сложности ИГ*. Пусть β — некоторая вершина ИГ. Предикат, определенный на множестве запросов, который принимает значение 1 на запросе x, если запрос проходит в вершину β, и 0 — в противном случае, назовем функцией фильтра вершины β и обозначим $\varphi_\beta(x)$.

Сложностью ИГ U на запросе $x \in X$ назовем число $T(U,x) = \sum_{\beta \in \mathcal{P}} \varphi_\beta(x) + \sum_{\beta \in \mathcal{R} \setminus \mathcal{P}} \psi_\beta \cdot \varphi_\beta(x)$, где $\mathcal{R}$ — множество вершин ИГ U, $\mathcal{P}$ — множество переключательных вершин ИГ U, ψ_β — количество ребер, исходящих из вершины β.

Скажем, что базовое множество $\mathcal{F}$ *измеримое*, если каждая функция из $\mathcal{F}$ — измеримая (относительно алгебры σ). Далее всюду будем предполагать, что базовое множество измеримое. В этом случае для любого ИГ U над $\mathcal{F}$ функция $T(U,x)$ как функция от x измерима.

Сложностью ИГ U назовем математическое ожидание величины $T(U,x)$, то есть число $T(U) = \mathbf{M}_x\, T(U,x)$. *Объемом* $Q(U)$ ИГ U назовем число ребер в графе U. *Сложностью задачи I при базовом множестве $\mathcal{F}$* назовем число $T(I,\mathcal{F}) = \inf\{T(U) : U \in \mathcal{U}(I,\mathcal{F})\}$, где $\mathcal{U}(I,\mathcal{F})$ — множество всех ИГ над базовым множеством $\mathcal{F}$, решающих ЗИП I.

Дадим описание типа задач поиска, который соответствует двумерной задаче интервального поиска.

Пусть $Y_{int2} = (0,1]^2$ — множество записей и

$$X_{int2} = \{x = (u_1, v_1, u_2, v_2) : 0 < u_i \le v_i \le 1,\ i = 1,2\} \text{ —}$$

множество запросов. Пусть на множестве X_{int2} задано вероятностное пространство $\langle X_{int2}, \sigma, \mathbf{P} \rangle$, где $\mathbf{P}$ задается функцией плотности вероятности $p(x)$.

Отношение поиска ρ_{int2} определено на $X_{int2} \times Y_{int2}$ и задается следующим соотношением:

$$(u_1, v_1, u_2, v_2)\rho_{int2}(y_1, y_2) \Longleftrightarrow u_i \le y_i \le v_i, \quad i = 1,2.$$

Тогда тип $S_{int2} = \langle X_{int2}, Y_{int2}, \rho_{int2}, \sigma, \mathbf{P}\rangle$ назовем типом двумерного интервального поиска.

Пусть $\mathbb{R}$ — вещественная прямая, а $\mathbb{R}^2$ — вещественная плоскость.

Если $a \in \mathbb{R}$, то через $]a[$ обозначим наименьшее целое, не меньшее чем a.

Пусть

$$F_1 = \{\chi_{a,\rho_{int2}} : a \in Y_{int2}\}, \tag{1}$$

$$G_1 = \{g_a(x) = \begin{cases} 1, & \text{если } x\rho_{int2}a \\ 2 & \text{в противном случае} \end{cases} : \ a \in Y_{int2}, \ x \in X_{int2}\}, \tag{2}$$

$$G_2 = \{g^1_{\cdot,m}(u_1, v_1, u_2, v_2) =]u_1 \cdot m[: m \in \mathbb{N}\}, \tag{3}$$

$$G_3 = \{g^1_{\leq,a}(u_1, v_1, u_2, v_2) = \begin{cases} 1, & \text{если } u_1 \leq a \\ 2, & \text{если } u_1 > a \end{cases} : \ a \in (0,1]\}, \tag{4}$$

$$G_4 = \{g^2_{\cdot,m}(u_1, v_1, u_2, v_2) =]v_1 \cdot m[: m \in \mathbb{N}\}, \tag{5}$$

$$G_5 = \{g^2_{\leq,a}(u_1, v_1, u_2, v_2) = \begin{cases} 1, & \text{если } v_1 \leq a \\ 2, & \text{если } v_1 > a \end{cases} : \ a \in (0,1]\}, \tag{6}$$

$$G_6 = \{g^3_{\cdot,m}(u_1, v_1, u_2, v_2) =]v_2 \cdot m[: m \in \mathbb{N}\}, \tag{7}$$

$$G_7 = \{g^3_{<,a}(u_1, v_1, u_2, v_2) = \begin{cases} 1, & \text{если } v_2 < a \\ 2, & \text{если } v_2 \geq a \end{cases} : \ a \in (0,1]\}, \tag{8}$$

$$\mathcal{F} = \langle \{1\} \cup F_1, G_1 \cup G_2 \cup G_3 \cup G_4 \cup G_5 \cup G_6 \cup G_7\rangle, \tag{9}$$

где 1 — функция тождественная единица, определенная на X_{int2}.

Теорема 1. *Если $I = \langle X_{int2}, V, \rho_{int2}\rangle$ — двумерная задача интервального поиска, где $|V| = k$, $\mathcal{F}$ — базовое множество, определяемое соотношениями (1)–(9), функция плотности вероятности $p(x)$, определяющая вероятностную меру $\mathbf{P}$, ограничена константой c, то*

$$\sum_{y \in V} \mathbf{P}(O(y, \rho_{int2})) \leq T(I, \mathcal{F}, \, k(c+9) + 2\sqrt{k}) \leq \sum_{y \in V} \mathbf{P}(O(y, \rho_{int2})) + 14\sqrt{k}.$$

Нижняя оценка следует из [3, Теорема 4] или [4, Теорема 3]. Смысл нижней оценки в том, что время поиска не может быть меньше времени, необходимого на перечисление ответа.

Здесь будет приведено доказательство верхней оценки.

3 Доказательство основного результата

На множестве $\mathbb{R}^2 \times \mathbb{R}^2$ введем отношение "$\leq$"следующим образом:

$$(x_1, y_1) \leq (x_2, y_2) \Longleftrightarrow x_1 \leq x_2 \& y_1 \leq y_2.$$

Скажем, что пара точек $(x_1, y_1), (x_2, y_2) \in \mathbb{R}^2$ *сравнима*, если $(x_1, y_1) \leq (x_2, y_2) \vee (x_2, y_2) \leq (x_1, y_1)$, в противном случае эта пара точек называется *несравнимой.*

Множество точек $S \subseteq \mathbb{R}^2$ назовем *поперечным (продольным) слоем*, если любая пара точек из S не сравнима (сравнима).

Множество точек $S \subseteq \mathbb{R}^2$ назовем *слоем*, если S поперечный или продольный слой.

Лемма 1. *Если $I = \langle X_{int2}, V, \rho_{int2} \rangle$ — двумерная задача интервального поиска, такая что V — слой мощности k, $\mathcal{F}$ — базовое множество, определяемое соотношениями (1)–(9), функция плотности вероятности $p(x)$, определяющая вероятностную меру $\mathbf{P}$, ограничена константой c, то существует ИГ $U_V \in \mathcal{U}(I, \mathcal{F})$ такой, что $Q(U_V) \leq k(c+9)$, $T(U_V) \leq 6 + \sum_{y \in V} \mathbf{P}(O(y, \rho_{int2}))$.*

Доказательство. Прежде, чем перейти к доказательству, дадим неформальное описание алгоритма, на котором достигается оценка леммы.

Пусть на вход алгоритма поступает запрос (u_1, v_1, u_2, v_2). В библиотеке, упорядоченной по возрастанию абсцисс точек из библиотеки, находится наименьшая по абсциссе точка, абсцисса которой не меньше, чем u_1. Эта задача соответствует первой задаче о близости, описанной, например, в [3, раздел 2.3]. Решать эту задачу будем методом, описанным в [3, Теорема 13]. Согласно этим теоремам ее можно решить за константное в среднем время с линейными затратами памяти.

Если найденная точка попадает в прямоугольник (u_1, v_1, u_2, v_2), то, начиная с найденной точки, просматриваем точки библиотеки в направлении возрастания абсцисс и включаем их в ответ пока просматриваемые точки попадают в прямоугольник (u_1, v_1, u_2, v_2). Процесс прекращается если очередная точка не попала в прямоугольник (u_1, v_1, u_2, v_2) или если просмотрены все точки библиотеки.

Если найденная точка не попадает в прямоугольник (u_1, v_1, u_2, v_2), то это не означает, что ответ пуст, просто она может не удовлетворить ограничениям по ординате. Предположим сначала, что библиотека является продольным слоем. Тогда в библиотеке, упорядоченной по возрастанию ординат точек из библиотеки, находится наименьшая по ординате точка, ордината которой не меньше, чем v_1. Это опять же первая задача о близости, и ее можно решить за константное в среднем время с линейными затратами памяти. Если найденная точка не попадает в прямоугольник (u_1, v_1, u_2, v_2), то ответ на запрос пуст. Если найденная точка попадает в прямоугольник (u_1, v_1, u_2, v_2), то, начиная с найденной точки, просматриваем точки библиотеки в направлении возрастания ординат и включаем их в ответ пока просматриваемые точки попадают в прямоугольник (u_1, v_1, u_2, v_2). Процесс прекращается если очередная точка не попала в прямоугольник (u_1, v_1, u_2, v_2) или если просмотрены все точки библиотеки.

В случае когда библиотека поперечный слой, в библиотеке, упорядоченной по возрастанию ординат точек из библиотеки, находится наибольшая по ординате точка, ордината которой не больше, чем v_2. Это вторая задача о близости, описанная в [3, раздел 2.3], и ее также можно решить за константное в среднем время с линейными затратами памяти. В этом случае, если найденная точка попадает в прямоугольник (u_1, v_1, u_2, v_2), то, начиная с найденной точки, просматриваем точки библиотеки в направлении убывания ординат и включаем их в ответ пока просматриваемые точки попадают в прямоугольник (u_1, v_1, u_2, v_2).

Как видно из описания предлагаемый алгоритм имеет линейные затраты по памяти и в среднем затраты по времени равны времени перечисления ответа плюс некоторая константа.

Перейдем к формальному доказательству теоремы.

Пусть $V = \{(y_1^1, y_1^2), (y_2^1, y_2^2), \dots, (y_k^1, y_k^2)\}$. Пусть $A = \{a_1, a_2, \dots, a_q\}$ — упорядоченное по возрастанию множество абсцисс точек из V (у точки (y_i^1, y_i^2) y_i^1 — абсцисса, y_i^2 — ордината). Если в множестве V все точки имеют различные абсциссы, то $q = k$, иначе $q < k$. Поскольку в поперечном слое и абсциссы, и ординаты всех точек различны, то $q < k$ может быть только в случае продольного слоя.

Пусть $m =]q \cdot c/2[$. Так как $c \geq 4$, то $m > q$.

Построим ИГ U_m^1, который будет решать первую задачу о близости для множества A.

Возьмем вершину β_0 и объявим ее корнем графа. Выпустим из β_0 m ребер, припишем им числа от 1 до m, объявим β_0 точкой переключения и припишем ей переключатель $g^1_{\cdot,m}$.

Пусть $A_i = ((i-1)/m, i/m] \cap A,\ l_i = |A_i|,\ i = \overline{1,m}$.

Конец ребра с номером i обозначим β_i.

Для всех i таких, что $V_i \neq \emptyset$, выполним следующую процедуру. Выпустим из вершины β_i бинарное сбалансированное дерево D_i с l_i+1 концевыми вершинами и высоты $]\log_2(l_i+1)[$. Здесь и далее бинарное дерево — это ориентированное дерево, в каждую вершину которого, кроме корня, входит одно ребро, и из каждой вершины которого, кроме концевых вершин, исходит два ребра; корень — это вершина, в которую не входит ни одно ребро; концевые вершины — это вершины из которых не исходит ни одно ребро; высота вершины — это длина пути от корня к данной вершине; сбалансированное дерево — это дерево, высота концевых вершин которого отличается не более чем на 1; высота дерева — это максимальная высота концевых вершин дерева.

Объявим все концевые вершины этого дерева D_i, кроме последней (самой правой), листьями и сопоставим им слева направо в порядке возрастания элементы из A_i. Здесь и далее будем представлять дерево в виде некоторой его укладки на плоскости и понимать направления "вправо", "влево"как направления на этой плоскости относительно данной укладки.

Для произвольной внутренней вершины β дерева D_i обозначим через V_β множество записей, соответствующих листьям, достижимым из β. Здесь и далее вершина α достижима из вершины β, если существует ориентированный путь ведущий из β в α. Пусть β' — вершина, в которую ведет левое ребро из β. Пусть

$$a_\beta = \max_{a \in V_{\beta'}} a.$$

Объявим все внутренние вершины дерева D_i вершинами переключения и для каждой внутренней вершины β левому ребру, из нее выходящему, припишем 1, а правому — 2, а самой вершине припишем переключатель $g^1_{\leq, a_\beta}$.

Пусть $i \in \{\overline{1,m}\}$. Обозначим $j(i)$ такой номер, что $j(i) > i$, $|A_{j(i)}| > 0$ и не существует j' такого, что $|A_{j'}| > 0$, $j' > i$ и $j' < j(i)$, то есть $j(i)$ — индекс ближайшего сверху непустого множества A_j. Если такого множества нет, то $j(i) = 0$.

Теперь для каждого дерева D_i самую правую концевую вершину дерева отождествим с самым левым листом дерева $D_{j(i)}$, если $j(i) \neq 0$.

Для каждого такого i, что $l_i = 0$, вершину β_i отождествим с самым левым листом дерева $D_{j(i)}$, если $j(i) \neq 0$.

Полученный граф и будет ИГ U_m^1.

Обозначим через α_i концевую вершину, которой приписана запись a_i ($i = 1, \ldots, q$), а через α_{q+1} самую правую концевую вершину в самом правом дереве D_i. Легко проверить, что функция проводимости между корнем и вершиной α_i ($i = 1, \ldots, q+1$) имеет вид:

$$f_i(u_1, v_1, u_2, v_2) = \begin{cases} 1, & \text{если } a_{i-1} < u_1 \leq a_i, \\ 0, & \text{в противном случае,} \end{cases}$$

где $a_0 = 0$, $a_{q+1} = 1$.

Подсчитаем объем графа U_m^1.

Поскольку каждое из D_i есть дерево, в котором полустепень исхода любой вершины равна 2, то в D_i будет ровно $2 \cdot (l_i + 1) - 2$ ребра. Отсюда следует, что

$$Q(U_m^1) = m + \sum_{i=1}^{m} (2\,(l_i + 1) - 2) = m + 2\sum_{i=1}^{m} l_i = m + 2\,k.$$

Подсчитаем сложность графа U_m^1.

Обозначим

$$L(u) = \begin{cases} u, & \text{если } 0 \leq u \leq 3 \\ \log_2(u+1) + 1, & \text{если } u \geq 3 \end{cases},$$

$$X_i = \{(u_1, v_1, u_2, v_2) \in X_{int2} : u_1 \in ((i-1)/m, i/m]\}.$$

Рассмотрим произвольный запрос $x = (u_1, v_1, u_2, v_2) \in X_{int2}$. Пусть $u_1 \in ((i-1)/m, i/m]$, т.е. $x \in X_i$. При $l_i = 0$ выполняется $T(U_m^1, x) = 1 = 1 + L(l_i)$.

Если $l_i > 0$, то запрос x выйдет к некоторой концевой вершине дерева D_i и при этом он пройдет не более $1 +]\log_2(l_i + 1)[$ переключательных вершин и, значит,

$$T(U_m^1, x) \leq 1 +]\log_2(l_i + 1)[\leq 1 + L(l_i).$$

По определению

$$\begin{aligned} T(U_m^1) &= \mathbf{M}_x\, T(U_m^1, x) = \int_{X_{int2}} T(U_m^1, x)\,\mathbf{P}(dx) = \\ &= \sum_{i=1}^{m} \int_{X_i} T(U_m^1, x)\,\mathbf{P}(dx) \le \sum_{i=1}^{m} (1 + L(l_i)) \cdot \mathbf{P}(X_i) = \\ &= 1 + \sum_{i=1}^{m} L(l_i) \cdot \mathbf{P}(X_i) \le 1 + \frac{c}{2m} \sum_{i=1}^{m} L(l_i). \end{aligned}$$

Так как $m > q$, то в силу вогнутости функции $L(u)$ и согласно лемме 14 из [3]

$$T(U_m^1) \le 1 + \frac{c}{2m} \sum_{i=1}^{m} L(l_i) \le 1 + q\,\frac{c}{2m} \le 2.$$

Пусть библиотека V упорядочена в порядке возрастания абсцисс точек, и если абсциссы точек совпадают, то в порядке возрастания ординат, т.е. $y_1^1 \le y_2^1 \le \cdots \le y_k^1$, и если $y_i^1 = y_{i+1}^1$, то $y_i^2 < y_{i+1}^2$. Пусть $y_{j_i} = (y_{j_i}^1, y_{j_i}^2) \in V$, такая точка, что $y_{j_i}^1 = a_i$ и $y_{j_i}^2$ минимально среди тех точек из V, для которых абсцисса равна a_i $(i = 1, \ldots, q)$. Ясно, что если V поперечный слой, то $j_i = i$.

Введем вершину γ_0. Она будет корнем той части информационного графа, в которой будет осуществляться решение задачи о близости по ординатам. Введем k вершин δ_j $(j = 1, \ldots, k)$, объявим эти вершины листьями ИГ и припишем листу δ_j запись $y_j = (y_j^1, y_j^2) \in V$.

Для каждого листа α_i $(i = 1, \ldots, q)$ проделаем следующее. Объявим вершину α_i обычной вершиной и уберем запись a_i с этой вершины; объявим α_i переключательной вершиной и припишем ей переключатель $g_{y_{j_i}}$; выпустим из α_i два ребра; первое ребро направим в вершину δ_{j_i} и припишем ему число 1; второе ребро направим в вершину γ_0 и припишем ему число 2.

Из каждого листа δ_j $(j = 1, \ldots, k-1)$ выпустим ребро в лист δ_{j+1}, объявим это ребро предикатным и припишем ему предикат $\chi_{y_{j+1}, \rho_{int2}}$.

Построенный на данный момент ИГ функционирует следующим образом. Если на корень ИГ подается запрос $x = (u_1, v_1, u_2, v_2)$, то он проходит в такую вершину α_i, что $y_{j_{i-1}} < u_1 \le y_{j_i}$ (здесь $y_{j_0} = 0$). Далее если $x \rho_{int2} y_{j_i}$, то y_{j_i} включается в ответ и по цепочке предикатных ребер осуществляется просмотр записей вдоль слоя, пока они удовлетворяют запросу. Если y_{j_i} не удовлетворяет запросу, т.е. y_{j_i} не попадает в прямоугольник (u_1, v_1, u_2, v_2), то запрос проходит в вершину γ_0.

Теперь из вершины γ_0 построим часть информационного графа, в которой будет осуществляться решение задачи о близости по ординатам.

Упорядочим библиотеку V в порядке возрастания ординат точек, и если ординаты точек совпадают, то в порядке возрастания абсцисс. Чтобы не использовать двойные и тройные индексы, будем считать, что после такого упорядочения библиотека имеет вид $V = \{(z_1^1, z_1^2), (z_2^1, z_2^2), \ldots, (z_k^1, z_k^2)\}$, т.е. $z_1^2 \leq z_2^2 \leq \cdots \leq z_k^2$, и если $z_i^2 = z_{i+1}^2$, то $z_i^1 < z_{i+1}^1$.

Сначала рассмотрим случай, когда V — продольный слой.

Пусть $B = \{b_1, b_2, \ldots, b_p\}$ — упорядоченное по возрастанию множество ординат точек из V $(p \leq k)$.

Пусть $n =]p \cdot c/2[$. Ясно, что $n > p$.

По аналогии с ИГ U_m^1 построим ИГ U_n^2, который будет решать первую задачу о близости для множества B, но только вместо переключателя $g^1_{\cdot,m}$ мы воспользуемся переключателем $g^2_{\cdot,n}$, а вместо переключателей $g^1_{\leq,a_\beta}$ — переключателями $g^2_{\leq,b_\beta}$.

Вершину, которой приписано b_i будем обозначать ζ_i $(i = 1, 2, \ldots, p)$, а через ζ_{p+1} самую правую концевую вершину в самом правом дереве D_i из U_n^2. Легко проверить, что функция проводимости между вершиной γ_0 и вершиной ζ_i $(i = 1, \ldots, p+1)$ имеет вид:

$$f_i'(u_1, v_1, u_2, v_2) = \begin{cases} 1, & \text{если } b_{i-1} < v_1 \leq b_i, \\ 0, & \text{в противном случае,} \end{cases}$$

где $b_0 = 0$, $b_{p+1} = 1$.

Также как и в случае с ИГ U_m^1 объем ИГ U_n^2 равен $Q(U_n^2) = n + 2\,k$, а сложность $T(U_n^2) \leq 2$.

Пусть $z_{j_i} = (z_{j_i}^1, z_{j_i}^2) \in V$, такая точка, что $z_{j_i}^2 = b_i$ и $z_{j_i}^1$ минимально среди тех точек из V, для которых ордината равна b_i $(i = 1, \ldots, p)$.

Введем k вершин η_j $(j = 1, \ldots, k)$, объявим эти вершины листьями ИГ и припишем листу η_j запись $z_j = (z_j^1, z_j^2) \in V$.

Для каждого листа ζ_i $(i = 1, \ldots, p)$ проделаем следующее. Объявим вершину ζ_i обычной вершиной и уберем запись b_i с этой вершины; выпустим из ζ_i ребро в вершину η_{j_i} и припишем ему предикат $\chi_{z_{j_i}, \rho_{int2}}$.

Из каждого листа η_j $(j = 1, \ldots, k-1)$ выпустим ребро в лист η_{j+1}, объявим это ребро предикатным и припишем ему предикат $\chi_{z_{j+1}, \rho_{int2}}$.

Полученный ИГ обозначим U_V. Он будет решать нашу задачу описанным выше методом в случае продольного слоя.

В случае когда V — поперечный слой, мы из вершины γ_0 построим ИГ для решения второй задачи о близости следующим образом.

Пусть $B = \{z_1^2, z_2^2, \ldots, z_k^2\}$, $n =]k \cdot c/2[$.

Выпустим из γ_0 n ребер, припишем им числа от 1 до n, объявим γ_0 точкой переключения и припишем ей переключатель $g^3_{\cdot,n}(u_1, v_1, u_2, v_2)$.

Пусть $B_i = ((i-1)/n, i/n] \cap V$, $l'_i = |B_i|$, $i = \overline{1,n}$.

Конец ребра с номером i обозначим γ_i.

Для всех i таких, что $B_i \neq \emptyset$, выполним следующую процедуру. Выпустим из вершины γ_i бинарное сбалансированное дерево D'_i с $l'_i + 1$ концевыми вершинами и высоты $]\log_2(l'_i + 1)[$.

Объявим все концевые вершины этого дерева D'_i, кроме первой (самой левой), листьями и сопоставим им слева направо в порядке возрастания элементы из B_i.

Для произвольной внутренней вершины β дерева D'_i обозначим

$$b_\beta = \min_{b \in V_{\beta''}} b,$$

где β'' — конец правого ребра, исходящего из β.

Объявим все внутренние вершины дерева D'_i вершинами переключения и для каждой внутренней вершины β левому ребру, из нее выходящему, припишем 1, а правому — 2, а самой вершине припишем переключатель $g^3_{<,b_\beta}(x)$.

Пусть $i \in \{\overline{1,n}\}$. Обозначим $j(i)$ такой номер, что $j(i) < i$, $|B_{j(i)}| > 0$ и не существует j' такого, что $|B_{j'}| > 0$, $j' < i$ и $j' > j(i)$, то есть $j(i)$ — индекс ближайшего снизу непустого множества B_j. Если такого множества нет, то $j(i) = 0$.

Теперь для каждого дерева D'_i самую левую концевую вершину дерева отождествим с самым правым листом дерева $D'_{j(i)}$, если $j(i) \neq 0$.

Для каждого такого i, что $l'_i = 0$, вершину γ_i отождествим с самым правым листом дерева $D'_{j(i)}$, если $j(i) \neq 0$.

Полученный граф с корнем в γ_0 будем обозначать U_n^2.

Вершину, которой приписано z_i^2 будем обозначать ζ_i $(i = 1, 2, \ldots, k)$, а через ζ_0 первую концевую вершину первого дерева D'_i из U_n^2. Легко проверить, что функция проводимости между вершиной γ_0 и вершиной

ζ_i $(i = 1, \ldots, k-1)$ имеет вид:

$$f_i''(u_1, v_1, u_2, v_2) = \begin{cases} 1, & \text{если } z_i^2 \le v_2 < z_{i+1}^2, \\ 0, & \text{в противном случае,} \end{cases}$$

$$f_0''(u_1, v_1, u_2, v_2) = \begin{cases} 1, & \text{если } 0 < v_2 < z_1^2, \\ 0, & \text{в противном случае,} \end{cases}$$

$$f_k''(u_1, v_1, u_2, v_2) = \begin{cases} 1, & \text{если } z_k^2 \le v_2 \le 1, \\ 0, & \text{в противном случае.} \end{cases}$$

По аналогии с ИГ U_m^1 объем ИГ U_n^2 равен $Q(U_n^2) = n + 2\,k$, а сложность $T(U_n^2) \le 2$.

Введем k вершин η_i $(i = 1, \ldots, k)$, объявим эти вершины листьями ИГ и припишем листу η_i запись $z_i = (z_i^1, z_i^2) \in V$.

Для каждой листа ζ_i $(i = 1, \ldots, k)$ проделаем следующее. Объявим вершину ζ_i обычной вершиной и уберем запись z_i^2 с этой вершины; выпустим из ζ_i ребро в вершину η_i и припишем ему предикат $\chi_{z_i, \rho_{int2}}$.

Из каждого листа η_i $(i = 2, \ldots, k)$ выпустим ребро в лист η_{i-1}, объявим это ребро предикатным и припишем ему предикат $\chi_{z_{i-1}, \rho_{int2}}$.

Полученный ИГ обозначим U_V. Он будет решать нашу задачу описанным выше методом в случае поперечного слоя.

Непосредственной проверкой легко убедиться в том, что для любого $i = 1, \ldots, k$ выполняется $\varphi_{\delta_i}(x) \vee \varphi_{\eta_{j_i}}(x) = \chi_{y_i, \rho_{int2}}$, где z_{j_i} есть переобозначение y_i. Следовательно, согласно теореме 1 из [3] ИГ U_V решает задачу $I = \langle X_{int2}, V, \rho_{int2} \rangle$.

Объем ИГ U_V равен

$$Q(U_V) = m + 2k + 2k + k - 1 + n + 2k + k + k - 1 \le k(c + 9).$$

Так как мы всегда выходим либо к вершинам группы δ_i, либо к вершинам группы η_i, то сложность ИГ U_V можно оценить следующим образом

$$T(U_V) \le T(U_m^1) + 1 + T(U_n^2) + 1 + \sum_{y \in V} \mathbf{P}(O(y, \rho_{int2})) \le 6 + \sum_{y \in V} \mathbf{P}(O(y, \rho_{int2})).$$

Тем самым лемма доказана.

Лемма 2. *В любом множестве из $k^2 + 1$ точки вещественной плоскости ($k \in \mathbb{N}$) всегда можно выделить слой мощности $k + 1$.*

Доказательство. Для точек $y_1, y_2 \in \mathbb{R}^2$ договоримся писать $y_1 < y_2$, если $y_1 \le y_2$ и $y_1 \ne y_2$; $y_1 \lhd y_2$, если y_1 и y_2 не сравнимы, и абсцисса точки y_1 меньше абсциссы точки y_2.

Значимой точкой слоя назовем точку с максимальной ординатой. Если таких точек несколько (а это возможно только для продольного слоя), то значимой будет точка с максимальной абсциссой. Иными словами, точка y^* — *значимая точка* слоя S, если для любого y из S, отличного от y^*, выполняется $y < y^*$ в случае продольного слоя, и $y^* \lhd y$ в случае поперечного.

Доказательство леммы будем вести индукцией по k.

Базис индукции. $k = 1$. Тогда $k^2 + 1 = 2$. Две точки всегда сравнимы или несравнимы, т.е. образуют слой.

Индуктивный переход. Пусть из любых $k^2 + 1$ точки можно выбрать слой мощности $k + 1$.

Рассмотрим множество V точек плоскости такое, что $|V| = (k+1)^2 + 1$. Рассмотрим множество $V_1 = V$. Так как $|V_1| \ge k^2 + 1$, то согласно предположению индукции в V_1 существует слой S_1 мощности $k+1$. Пусть точка y_1 значимая точка для слоя S_1. Определим множество $V_2 = V_1 \setminus \{y_1\}$. Так как $|V_2| \ge k^2 + 1$, то согласно предположению индукции в V_2 существует слой S_2 мощности $k + 1$. Пусть точка y_2 значимая точка для слоя S_2. Далее переходим к множеству $V_3 = V_2 \setminus \{y_2\}$ и т.д. Продолжая таким образом, мы на $(2k+2)$-м шаге получим множество $V_{2k+2} = V_{2k+1} \setminus \{y_{2k+1}\} = V \setminus \{y_1, y_2, \ldots, y_{2k+1}\}$. $|V_{2k+2}| = (k+1)^2 + 1 - 2k - 1 = k^2 + 1$. По предположению индукции в V_{2k+1} существует слой S_{2k+2} мощности $k+1$, пусть точка y_{2k+2} значимая точка этого слоя.

Рассмотрим множество $V' = \{y_1, y_2, \ldots, y_{2k+2}\}$. Разобьем его на два подмножества V'_1 и V'_2 так, что V'_1 содержит значимые точки продольных слоев, а V'_2 — значимые точки поперечных.

Возможны следующие 5 ситуаций.

1) $|V'_1| \ge k + 1$ и в V'_1 существует пара y_{l_1}, y_{l_2} сравнимых точек. Пусть $y_{l_1} < y_{l_2}$. Так как y_{l_1} значимая точка слоя S_{l_1}, то $y_{l_2} \notin S_{l_1}$. Поскольку отношение "$\le$"ассоциативно, то множество $S = S_{l_1} \cup \{y_{l_2}\}$ образует продольный слой и $|S| = k + 2$.

2) $|V'_2| \ge k + 1$ и в V'_2 существует пара несравнимых точек y_{j_1}, y_{j_2} . Пусть $y_{j_1} \lhd y_{j_2}$. Так как y_{j_2} значимая точка слоя S_{j_2}, то абсцисса точки y_{j_1} меньше абсциссы любой из точек из S_{j_2}, а ордината точки y_{j_1} больше ординаты любой из точек из S_{j_2}), и $y_{j_1} \notin S_{j_2}$. Следовательно, множество $S' = \{y_{j_1}\} \cup S_{j_2}$ образует поперечный слой мощности $k + 2$.

3) $|V_1'| > k+1$ и в V_1' все точки попарно несравнимы. Тогда V_1' образует поперечный слой мощности не менее $k+2$.

4) $|V_2'| > k+1$ и в V_2' все точки попарно сравнимы. Тогда V_2' образует продольный слой мощности не менее $k+2$.

5) $|V_1'| = |V_2'| = k+1$ и в V_1' все точки попарно несравнимы, а в V_2' все точки попарно сравнимы, т.е. V_1' образует поперечный слой, а V_2' — продольный. Пусть y_p — значимая точка V_1', y_q — значимая точка V_2'. Возможны следующие 4 подслучая.

а) $y_p < y_q$. Тогда множество $S_p \cup \{y_q\}$ образует продольный слой мощности $k+2$.

б) $y_q < y_p$. Тогда множество $V_2' \cup \{y_p\}$ образует продольный слой мощности $k+2$.

в) $y_p \lhd y_q$. Тогда множество $\{y_p\} \cup S_q$ образует поперечный слой мощности $k+2$.

г) $y_q \lhd y_p$. Тогда множество $\{y_q\} \cup V_1'$ образует поперечный слой мощности $k+2$.

Тем самым, мы показали, что всегда можно найти слой мощности не менее, чем $k+2$, и доказали лемму.

Следствие 1. *В множестве из k точек вещественной плоскости всегда можно выделить слой мощности* $\left]\sqrt{k}\right[$.

Рассмотрим следующую вещественнозначную функцию натурального аргумента, задаваемую рекурсивно:

$$\begin{aligned} f(1) &= \sqrt{k}, \\ f(i) &= \sqrt{k - \sum_{j=1}^{i-1} f(j)}, \end{aligned}$$

где k — некоторая натуральная константа.

Докажем некоторые свойства этой функции.

Свойство 1. *Для любого $i > 1$ из области определения функции f справедливо $f(i-1) - f(i) > 1/2$.*

Доказательство. Согласно определению

$$\begin{aligned} f^2(i) &= k - \sum_{j=1}^{i-1} f(j), \\ f^2(i-1) &= k - \sum_{j=1}^{i-2} f(j). \end{aligned}$$

Вычитая эти равенства имеем

$$\begin{aligned} f^2(i) - f^2(i-1) &= -f(i-1), \\ f^2(i) &= (f(i-1) - 1/2)^2 - 1/4, \\ f^2(i) &< (f(i-1) - 1/2)^2, \\ f(i) &< f(i-1) - 1/2, \\ f(i-1) - f(i) &> 1/2, \end{aligned}$$

что и требовалось доказать.

Следствие 2. *Функция $f(i)$ — строго убывающая.*

Свойство 2. *Областью определения функции $f(i)$ является начальный отрезок натурального ряда $\{1, 2, \ldots, t_f\}$, где $t_f < 2\sqrt{k} + 1$.*

Доказательство. Из того, что функция $f(i)$ неотрицательная и строго убывающая с шагом не менее, чем $1/2$, следует, что область определения функции является начальным отрезком натурального ряда. Обозначим через t_f максимальное число из области определения. Из свойства 1 следует, что $f(i) < f(1) - (i-1)/2 = \sqrt{k} - (i-1)/2$. Поскольку $f(2\sqrt{k}+1) < \sqrt{k} - 2\sqrt{k}/2 = 0$, то $t_f < 2\sqrt{k}+1$, что и требовалось доказать.

Рассмотрим следующую функцию натурального аргумента, принимающую натуральные значения и задаваемую рекурсивно:

$$\begin{aligned} g(1) &= \left]\sqrt{k}\right[, \\ g(i) &= \left]\sqrt{k - \textstyle\sum_{j=1}^{i-1} g(j)}\right[, \end{aligned}$$

где k — некоторая натуральная константа.

Докажем некоторые свойства этой функции.

Свойство 3. *$g(i)$ — невозрастающая функция, и областью ее определения является начальный отрезок натурального ряда.*

Доказательство. Поскольку для любого i из области определения $g(i) \geq 1$ (функция g принимает натуральные значения), то функция $k - \sum_{j=1}^{i-1} g(j)$ строго убывающая с шагом не менее 1. Откуда сразу следует, что $g(i)$ — невозрастающая функция и область ее определения есть начальный отрезок натурального ряда, что и требовалось доказать.

Обозначим через t_g максимальное число из области определения функции $g(i)$.

Свойство 4. $\sum_{j=0}^{t_g} g(j) = k$.

Доказательство. Пусть $\sum_{j=0}^{t_g-1} g(j) = a$. Так как в точке t_g функция g определена и $g(t_g) \geq 1$, то $k - a \geq 1$ и $g(t_g) = \left]\sqrt{k-a}\right[\leq k - a$, причем равенство возможно только при $k - a \in \{1, 2\}$. Следовательно, $\sum_{j=1}^{t_g} g(j) = a + g(t_g) \leq a + k - a = k$. Если предположить, что $\sum_{j=1}^{t_g} g(j) < k$, то $k - \sum_{j=1}^{t_g} g(j) \geq 1$ и, следовательно, определено значение $\left]\sqrt{k - \sum_{j=1}^{t_g} g(j)}\right[$, что противоречит максимальности t_g.

Тем самым свойство доказано.

Свойство 5. *Для любого натурального $n \leq t_g$, выполняется $n \leq t_f$ и $\sum_{i=1}^{n} g(i) \geq \sum_{i=1}^{n} f(i)$.*

Доказательство будем вести индукцией по n.

Базис индукции. $n = 1$. $g(1) =]\sqrt{k}[\geq f(1) = \sqrt{k}$.

Индуктивный переход. Рассмотрим $n \leq t_g$. Сделаем индуктивное предположение $n - 1 \leq t_f$ и $\sum_{i=1}^{n-1} g(i) \geq \sum_{i=1}^{n-1} f(i)$.

Обозначим $a = \sum_{i=1}^{n-1} f(i)$.

Если $\sum_{i=1}^{n-1} g(i) = a$, то $g(n) =]\sqrt{k-a}[\geq \sqrt{k-a} = f(n)$. Следовательно, $n \leq t_f$ и $\sum_{i=1}^{n} g(i) \geq \sum_{i=1}^{n} f(i)$.

Пусть теперь $\sum_{i=1}^{n-1} g(i) = a + b$, где $b > 0$. Так как $n \leq t_g$, то $k - a - b \geq 1$ и $g(n) =]\sqrt{k-a-b}[$. Тогда $k - a \geq 1 + b > 1$ и $f(n) = \sqrt{k-a}$ определено, т.е. $n \leq t_f$.

$$\sum_{i=1}^{n} g(i) = a + b +]\sqrt{k-a-b}[\geq a +]\sqrt{k-a}[\geq \sum_{j=1}^{n} f(j).$$

Здесь мы воспользовались неравенством $c + \sqrt{d-c} \geq \sqrt{d}$, которое выполняется для любого $d \geq 1$ и любого $c \in [0, d]$. В самом деле, при возведении в квадрат обеих частей неравенства имеем $c^2 + 2c\sqrt{d-c} + d - c \geq d$. При $c \geq 1$ это неравенство очевидно. Рассмотрим случай, когда $c < 1$.

$$\begin{aligned} c(c + 2\sqrt{d-c} - 1) &\geq 0, \\ 2\sqrt{d-c} &\geq 1 - c, \\ 4(d-c) &\geq (1-c)^2, \end{aligned}$$

что очевидно при $d \geq 1$.

Тем самым свойство доказано.

Свойство 6. *$t_g < 2\sqrt{k}$.*

Доказательство. Согласно свойству 4 $\sum_{j=0}^{t_g} g(j) = k$. Согласно свойству 5 $\sum_{i=1}^{t_g} g(i) \geq \sum_{i=1}^{t_g} f(i)$. Следовательно, $k - \sum_{i=1}^{t_g} f(i) \geq 0$ и $f(t_g + 1) = \sqrt{k - \sum_{i=1}^{t_g} f(i)}$ определено. Откуда, вспоминая свойство 2, имеем $t_g \leq t_f - 1 < 2\sqrt{k}$, что и требовалось доказать.

Лемма 3. *Любое множество из k точек вещественной плоскости можно представить в виде прямой суммы менее, чем $2\sqrt{k}$ слоев.*

Доказательство. Пусть $V \subseteq \mathbb{R}^2$ такое, что $|V| = k$. Согласно следствию из леммы 2 в множестве $V_1 = V$ можно выделить слой S_1 такой, что $|S_1| =]\sqrt{k}[= g(1)$. В оставшемся множестве $V_2 = V_1 \setminus S_1$ мощности $k - g(1)$ можно выделить слой S_2 мощности $\left]\sqrt{k - g(1)}\right[= g(2)$. Далее переходим к рассмотрению множества $V_3 = V_2 \setminus S_2$, в котором есть слой S_3 мощности $\left]\sqrt{k - \sum_{i=1}^{2} g(i)}\right[= g(3)$ и т.д. пока не исчерпаем множество V. Понятно, что мощность i-го слоя будет равна $g(i)$ и согласно свойству 4 число слоев будет равно t_g. Вспоминая свойство 6, получаем утверждение леммы.

Доказательство теоремы. Пусть дана двумерная задача интервального поиска $I = \langle X_{int2}, V, \rho_{int2} \rangle$, где $|V| = k$. Согласно лемме 3 множество V можно представить в виде прямой суммы t_g слоев $V = \bigcup_{i=1}^{t_g} S_j$. Предлагаемый алгоритм решения этой задачи состоит в том, чтобы

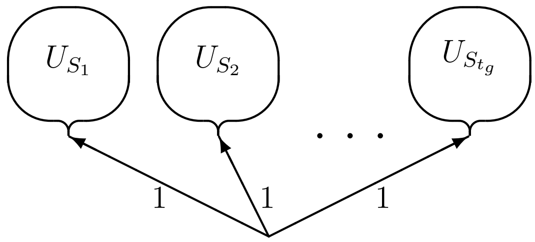

Рис. 1: ИГ U

для каждого слоя независимо воспользоваться алгоритмом, описанным в лемме 1.

Этому алгоритму соответствует ИГ U, изображенный на рисунке 1, где U_{S_i} — ИГ, описанный в лемме 1 и решающий ЗИП $\langle X_{int2}, S_i, \rho_{int2}\rangle$.

Очевидно, что данный ИГ U решает ЗИП I.

$$\begin{aligned}
Q(U) &= t_g + \sum_{i=1}^{t_g} Q(U_{S_i}) \leq t_g + \sum_{i=1}^{t_g} |S_i|(c+9) < k(c+9) + 2\sqrt{k},\\
T(U) &= t_g + \sum_{i=1}^{t_g} T(U_{S_i}) \leq 7t_g + \sum_{y\in V} \mathbf{P}(O(y, \rho_{int2})) < 14\sqrt{k} + \sum_{y\in V} \mathbf{P}(O(y, \rho_{int2})).
\end{aligned}$$

Тем самым, теорема 1 доказана.

Список литературы

[1] Гасанов Э. Э., Кузнецова И. В. О функциональной сложности двумерной задачи интервального поиска. *Дискретная математика* (2002) **14**, № 1.

[2] Гасанов Э. Э., Ерохин А.Н. Линейный по памяти непереборный алгоритм решения двумерной задачи интервального поиска. *Дискретная математика* (2004) **16**, № 4, 49–64.

[3] Гасанов Э. Э, Кудрявцев В. Б. *Теория хранения и поиска информации.* — М.: ФИЗМАТЛИТ, 2002.

[4] Гасанов Э. Э. Об одной математической модели информационного поиска. *Дискретная математика* (1991) **3**, № 2, 69–76.