

Математические основы цифровой обработки сигналов (Д. Н. Бабин, И. Л. Мазуренко)

Краткое описание практикума

Практикум по курсу “Математические основы цифровой обработки сигналов” содержит ряд теоретико-программистских практических упражнений, посвященных цифровой обработке и распознаванию сигналов, изображений и видеозаписей. Выполнение большинства упражнений производится в одном из компьютерных математических прикладных пакетов. В рамках практикума предусмотрены также практические упражнения по созданию оптимизированных по временной сложности реализации программ на языке С на реальных процессорах цифровой обработки сигналов.

1. Дискретизация сигнала, преобразование Фурье

1.1. Изменить частоту дискретизации сигнала. На вход программы, реализованной в виде m-функции, поступает входной вектор-сигнал x , а также входная F_1 и выходная F_2 частоты дискретизации в Герцах. Программа должна изменить частоту дискретизации сигнала x и вернуть модифицированный сигнал y длины $F_2|x|/F_1$. Задачу необходимо решить путем создания алгоритма восстановления отсчетов выходного сигнала по теореме Котельникова.

1.2. Алгоритм быстрого преобразования Фурье с прореживанием по времени. Должен быть реализован в виде m-функции, которая для цифрового сигнала x длины 2^N возвращает $1 + 2^{N-1}$ комплексных коэффициентов Фурье.

1.3. Алгоритм быстрого преобразования Фурье с прореживанием по частоте. Должен быть реализован в виде m-функции, которая для цифрового сигнала x длины 2^N возвращает $1 + 2^{N-1}$ комплексных коэффициентов Фурье.

1.4. Двоично-инверсный алгоритм перестановки входной последова-

тельности. Должен быть реализован в виде m -функции, которая для цифрового сигнала x длины 2^N осуществляет перестановку его элементов согласно двоично-инверсному алгоритму.

1.5. Период основного тона речевого сигнала. Реализовать простейший алгоритм оценки зависимости от времени периода основного тона речевого сигнала путем поиска точки максимума автокорреляции сигнала на окне анализа. Частота дискретизации сигнала, размер и шаг окна анализа и другие пороговые параметры определяются эвристически.

2. Цифровая фильтрация сигналов

2.1. Фильтр с конечной импульсной характеристикой. Реализовать в среде m -функцию фильтрации сигнала фильтром с КИХ. Входные параметры функции — вектор коэффициентов фильтра, входной вектор-сигнал. Выход функции — результат фильтрации.

2.2. Фильтр с бесконечной импульсной характеристикой. Реализовать в среде m -функцию фильтрации сигнала БИХ-фильтром. Входные параметры функции — вектор коэффициентов b_i числителя передаточной функции $H(z)$ фильтра, вектор коэффициентов a_k знаменателя передаточной функции (кроме свободного члена, равного 1), входной вектор-сигнал. Выход функции — результат фильтрации.
$$H(z) = \frac{\sum_{i=0}^P b_i z^{-i}}{1 + \sum_{k=1}^Q a_k z^{-k}}.$$

2.3. Фильтр с нулевым фазовым искажением. Реализовать m -функцию фильтрации сигнала фильтром с нулевым фазовым искажением. Входные параметры функции — вектор коэффициентов фильтра, вектор коэффициентов a_k знаменателя передаточной функции (кроме свободного члена, равного 1), входной вектор-сигнал. Выход функции — результат фильтрации. Фильтр требуется реализовать на основе идеи двухпроходной фильтрации.

2.4. Фильтрация сигнала, основанная на применении БПФ. Реализовать в m -функцию фильтрации сигнала на основе применения прямого и обратного быстрого преобразования Фурье. Входные параметры функции — вектор комплексных коэффициентов фильтра, задающих искажение сигнала в спектральной области, входной вектор-сигнал. Выход функции — результат фильтрации.

2.5. Обобщенный метод Ньютона адаптивной фильтрации сигналов. Реализовать обобщенный метод Ньютона адаптивной фильтрации в виде m -функции. Вход функции: очередной отсчет входного сигнала x , очередной отсчет опорного сигнала d , величина ошибки предсказания предыдущего отсчета сигнала e , текущая оценка вектора коэффициентов фильтра h . Выход функции — результат предсказания y , новая оценка вектора коэффициентов фильтра h .

2.6. Метод наименьших квадратов адаптивной фильтрации сигнала (LMS). Реализовать метод наименьших квадратов адаптивной фильтрации сиг-

нала в виде m -функции. Функция должна быть реализована со сложностью $2L + C$ умножений, где L — длина фильтра. Вход функции: очередной отсчет входного сигнала x , очередной отсчет опорного сигнала d , величина ошибки предсказания предыдущего отсчета сигнала e , текущая оценка вектора коэффициентов фильтра h . Выход функции — результат предсказания y , новая оценка вектора коэффициентов фильтра h .

2.7. Нормализованный метод наименьших квадратов адаптивной фильтрации сигнала (NLMS). Реализовать метод наименьших квадратов адаптивной фильтрации сигнала в виде m -функции. Функция должна быть реализована со сложностью $2L + C$ умножений, где L — длина фильтра. Вход функции: очередной отсчет входного сигнала x , очередной отсчет опорного сигнала d , величина ошибки предсказания предыдущего отсчета сигнала e , текущая оценка вектора коэффициентов фильтра h . Выход функции — результат предсказания y , новая оценка вектора коэффициентов фильтра h .

2.8. Рекурсивный метод наименьших квадратов адаптивной фильтрации сигнала (RLS). Реализовать рекурсивный метод наименьших квадратов адаптивной фильтрации сигнала в виде m -функции. Вход функции: очередной отсчет входного сигнала x , очередной отсчет опорного сигнала d , величина ошибки предсказания предыдущего отсчета сигнала e , текущая оценка вектора коэффициентов фильтра h . Выход функции — результат предсказания y , новая оценка вектора коэффициентов фильтра h .

2.9.* Быстрый рекурсивный метод наименьших квадратов адаптивной фильтрации сигналов (FRLS). Реализовать быстрый рекурсивный метод наименьших квадратов адаптивной фильтрации сигналов в виде m -функции. Функция должна быть реализована со сложностью $10L + C$ умножений, где L — длина фильтра. Вход функции: очередной отсчет входного сигнала x , очередной отсчет опорного сигнала d , величина ошибки предсказания предыдущего отсчета сигнала e , текущая оценка вектора коэффициентов фильтра h . Выход функции — результат предсказания y , новая оценка вектора коэффициентов фильтра h .

2.10. Метод аффинных проекций адаптивной фильтрации сигнала (АРА). Реализовать метод аффинных проекций адаптивной фильтрации сигнала в виде m -функции. Вход функции: очередной отсчет входного сигнала x , очередной отсчет опорного сигнала d , величина ошибки предсказания предыдущего отсчета сигнала e , текущая оценка вектора коэффициентов фильтра h . Выход функции — результат предсказания y , новая оценка вектора коэффициентов фильтра h .

2.11.* Быстрый метод аффинных проекций адаптивной фильтрации сигнала (FAP). Реализовать быстрый метод аффинных проекций адаптивной фильтрации сигнала в виде m -функции. Функция должна быть реализована со сложностью $2L + 10N + C$ умножений, где L — длина фильтра, N — размер аффинной проекции. Вход функции: очередной отсчет входного сигнала x , очередной отсчет опорного сигнала d , величина ошибки предсказания предыдущего

го отсчета сигнала e , текущая оценка вектора коэффициентов фильтра h . Выход функции — результат предсказания y , новая оценка вектора коэффициентов фильтра h .

2.12.* **Пропорциональный нормализованный метод наименьших квадратов адаптивной фильтрации (PNLMS)**. Реализовать пропорциональный нормализованный метод наименьших квадратов адаптивной фильтрации сигнала в виде m -функции. Вход функции: очередной отсчет входного сигнала x , очередной отсчет опорного сигнала d , величина ошибки предсказания предыдущего отсчета сигнала e , текущая оценка вектора коэффициентов фильтра h . Выход функции — результат предсказания y , новая оценка вектора коэффициентов фильтра h .

3. Сжатие речи

3.1. Реализовать в виде m -функции алгоритм Левинсона-Дурбина решения системы уравнений Юла-Уолкера.

3.2. Реализовать обобщенный алгоритм Левинсона решения системы уравнений с Теплицевой матрицей.

3.3. Реализовать алгоритм поиска коэффициентов линейного предсказания и коэффициентов отражения для заданного сигнала x длины L и числа коэффициентов N . Выход функции — вектор коэффициентов линейного предсказания, вектор коэффициентов отражения.

3.4. Реализовать алгоритм преобразования коэффициентов линейного предсказания в линейные спектральные пары.

3.5. Запрограммировать кодер / декодер звукового сигнала согласно стандарту G.711 (А-закон).

3.6. Запрограммировать кодер / декодер звукового сигнала согласно стандарту G.711 (μ -закон)

3.7.* Реализовать детектор активности речи, описанный в стандарте G.729B

3.8.* Реализовать генератор комфортного шума, описанный в стандарте G.729B

3.9. Написать функцию, реализующую кодирование / декодирование сигналов с помощью алгоритма DPCM

3.10. Написать функцию, реализующую кодирование / декодирование сигналов с помощью алгоритма ADPCM

3.11.* Реализовать алгоритм кодирования речевых сигналов, работающих согласно стандарту G.729

3.12.* Реализовать алгоритм декодирования речевых сигналов, работающих согласно стандарту G.729

4. Улучшение качества речи

- 4.1. Реализовать алгоритм восстановления потерянных пакетов (PLC), описанный в стандарте G.711, Appendix I
- 4.2.* Реализовать алгоритм адаптивной шумоочистки, описанный в стандарте EVRC, см. http://www.3gpp2.org/public_html/specs/C.S0014-A_v1.0_040426.pdf
- 4.3.* Используя реализацию алгоритма определения речевой активности, реализовать алгоритм оценки амплитудных параметров шума.
- 4.4. Реализовать алгоритм генерации комфортного шума (CNG), описанный в стандарте G.729
- 4.5. Реализовать алгоритм автоматического контроля уровня входного сигнала (ALC/AGC) согласно стандарту G.169.
- 4.6.* Реализовать алгоритм компенсации высокого уровня сигнала (HLC).
- 4.7.* Реализовать алгоритм автоматического подавления электрического эха согласно стандарту G.168, работающий на основе алгоритма NLMS. Автоматический контроль акустического эха

5. Сжатие изображений и видеосигналов, транскодирование видео

- 5.1. Реализовать алгоритм одномерного прямого и обратного дискретного косинусного преобразования вектора заданной длины
- 5.2. Реализовать алгоритм двумерного дискретного косинусного преобразования изображения размером $n \times n$ точек
- 5.3. Реализовать алгоритм линейного квантования 8×8 -матриц из стандарта JPEG, имеющий в качестве входной переменной параметр потери данных p
- 5.4. Написать функцию преобразования цветовых координат точки $RGB \leftrightarrow YUV$
- 5.5.* Реализовать алгоритм сжатия изображений JPEG (baseline profile)
- 5.6. Создать подпрограмму изменения размера изображения, изменяющую ширину и высоту кадра в $N : M$ раз для любых натуральных N и M
 - a. Методом бикубических сплайнов (двухпроходный алгоритм)
 - b. Методом бикубических сплайнов (однопроходный алгоритм)
 - c. Билинейным методом
 - d. Методом “ближнего соседа”

6. Распознавание речи

- 6.1.* Реализовать метод распознавания речи на основе алгоритма динамического искажения времени
- 6.2. Реализовать алгоритм Витерби приближенного вычисления условной вероятности наблюдения заданного слова на выходе заданной марковской модели.
- 6.3. Реализовать алгоритм прямого-обратного хода вычисления условной вероятности наблюдения заданного слова на выходе заданной марковской модели.
- 6.4. Реализовать алгоритм Баума-Уэлча настройки параметров скрытой марковской модели на заданной обучающей выборке.
- 6.5.* Реализовать метод распознавания речи на основе алгоритма Скрытых марковских моделей.

Рекомендуемая литература

- 1. Рабинер Л.Р., Голд Б. Теория и практика цифровой обработки сигналов.
- 2. Рабинер Л.Р., Шафер Р.В. Цифровая обработка речевых сигналов.
- 3. Блейхут Р. Быстрые алгоритмы цифровой обработки сигналов.
- 4. Simon Haykin. Adaptive filter theory.
- 5. Винцюк Т.К. Анализ, распознавание и интерпретация речевых сигналов. Киев: Наукова думка, 1987.
- 6. Винцюк Т.К. Сравнение ИКДП- и НММ - методов распознавания речи // Методы и средства информ. речи. Киев, 1991.
- 7. Рабинер Л. Р. Скрытые марковские модели и их применение в избранных приложениях при распознавании речи.
- 8. Рекомендации ITU G.711, G.729A,B, G.169, G.169 с сайта <http://www.itu.int/ITU-T/>.