

Министерство образования Республики Беларусь
Учреждение образования
“Белорусский государственный университет
информатики и радиоэлектроники”

Кафедра радиотехнических систем

С.Б. Саломатин, А.И. Бурак

**КОМПЬЮТЕРНОЕ МОДЕЛИРОВАНИЕ И ОБРАБОТКА
НЕСТАЦИОНАРНЫХ СИГНАЛОВ**

Методическое пособие к лабораторной работе по курсу
“Цифровая обработка сигналов”
для студентов специальностей “Радиотехника”, “Радиоэлектронные системы”
всех форм обучения

Минск 2004

УДК 621.391.23 (075.8)

ББК 32.811.3 я 73

С 60

Саломатин С.Б.

С 60 Компьютерное моделирование и обработка нестационарных сигналов: Метод. пособие к лабораторной работе по курсу “Цифровая обработка сигналов” для студ. спец. “Радиотехника”, “Радиоэлектронные системы” всех форм обуч./С.Б. Саломатин, А.И. Бурак. -Мн.: БГУИР, 2004. – 36 с.: ил.
ISBN 985-444-641-7

В методическом пособии рассмотрены алгоритмы и возможности моделирования адаптивных систем в пакете MatLab, приводятся содержание и порядок выполнения лабораторной работы по исследованию алгоритма идентификации канала связи.

УДК 621.391.23 (075.8)
ББК 32.811.3 я 73

ISBN 985-444-641-7

© Саломатин С.Б., Бурак А.И., 2004
© БГУИР, 2004

Содержание

ВВЕДЕНИЕ

1.ОСНОВНАЯ ИДЕЯ АДАПТИВНОЙ ОБРАБОТКИ СИГНАЛА

2.ОПТИМАЛЬНЫЙ ФИЛЬТР ВИНЕРА

3.АЛГОРИТМ LMS

4.АДАПТИВНАЯ ФИЛЬТРАЦИЯ НЕСТАЦИОНАРНОГО СИГНАЛА

5.АЛГОРИТМ RLS

6.АЛГОРИТМ КАЛМАНА

7.ПРИМЕНЕНИЕ АДАПТИВНЫХ ФИЛЬТРОВ

7.1.Идентификация систем

7.2.Подавление шума

7.3.Выравнивание характеристик канала связи

7.4.Эхоподавление

8.ФУНКЦИИ АДАПТИВНОЙ ФИЛЬТРАЦИИ В ПАКЕТЕ MATLAB

9.БЛОКИ АДАПТИВНОЙ ФИЛЬТРАЦИИ ИЗ НАБОРА DSP BLOKSET

10. СОДЕРЖАНИЕ И ПОРЯДОК ВЫПОЛНЕНИЯ ЛАБОРАТОРНОЙ РАБОТЫ

10.1.Программное обеспечение лабораторной работы

10.2.Содержание лабораторной работы

10.3.Порядок выполнения лабораторной работы

ЛИТЕРАТУРА

ВВЕДЕНИЕ

Цифровая обработка сигналов в сложных радиоэлектронных системах характеризуется рядом особенностей, осложняющих решение радиотехнических задач. В высококачественных системах цифровая обработка сигналов часто ведется в условиях *неопределенности* системных характеристик. Априорная и текущая информация о параметрах системы является неполной. Кроме того, при эксплуатации системы параметры её объектов и среды функционирования могут изменяться непредвиденным заранее образом и тогда *нестационарность* выступает как один из видов неопределенности. Стохастичность поведения также является важной чертой, характеризующей обработку сигналов в сложных системах. Это обусловлено наличием источников случайных помех и всякого рода второстепенных (с точки зрения решения задачи) процессов с непредсказуемым поведением. Таким образом, стохастичность и нестационарность объектов и условий их функционирования определяют фактор сложности системы.

Одним из наиболее перспективных путей преодоления трудностей, порождаемых этим фактором сложности, является применение методов адаптации. При адаптивной цифровой обработке информация об объекте и внешних воздействиях собирается во время работы устройства и используется для изменения параметров блоков обработки. Адаптивные устройства обработки сигнала действуют по принципу замкнутого контура. Входной сигнал фильтруется или взвешивается в программируемом фильтре для получения выходного сигнала, который затем сравнивается с полезным, стандартным или обучающим сигналом для нахождения сигнала ошибки. Затем этот сигнал ошибки используется для корректировки весовых параметров процессора обработки с целью постепенной минимизации ошибки. При этом адаптивной системе требуется минимальный объем исходной информации о поступающем сигнале.

Процессоры радиотехнической обработки можно разделить на два больших класса: адаптивные фильтры и адаптивные антенны. Адаптивные антенны осуще-

ствляют пространственную обработку сигнала с помощью антенной решетки, формируя диаграмму направленности таким образом, чтобы в ней создавался главный максимум в направлении прихода сигнала и происходила генерация нулей в направлении источников помех. В адаптивных цифровых фильтрах используется программируемый фильтр, частотная характеристика или передаточная функция которого изменяется (адаптируется) таким образом, чтобы пропустить без искажения полезные составляющие сигнала и ослабить нежелательные сигналы или помехи, т.е. уменьшить искажения входного полезного сигнала. Адаптивный фильтр действует по принципу оценивания статистических параметров поступающего сигнала и подстройки собственной переходной характеристики таким образом, чтобы минимизировать некоторую функцию стоимости. Такие адаптивные фильтры часто используются для восстановления на выходах каналов сигналов с меняющимися во времени характеристиками.

1. ОСНОВНАЯ ИДЕЯ АДАПТИВНОЙ ОБРАБОТКИ СИГНАЛА

Общая структура адаптивного фильтра показана на рис. 1. Входной дискретный сигнал $x(k)$ обрабатывается дискретным фильтром, в результате чего получается выходной сигнал $y(k)$. Этот выходной сигнал сравнивается с *образцовым* сигналом $d(k)$, разность между ними образует сигнал ошибки $e(k)$. Назначение адаптивного фильтра – минимизировать ошибку воспроизведения образцового сигнала. С этой целью блок адаптации после обработки каждого отсчета анализирует сигнал ошибки и дополнительные данные, поступающие из фильтра, используя результаты этого анализа для подстройки параметров (коэффициентов) фильтра.

Возможен и иной вариант адаптации, при котором образцовый сигнал не используется или требуемые параметры сигнала напрямую не могут быть измерены. Такой режим работы называется *адаптацией по косвенным данным* (blind adaptation) или *обучением без учителя* (unsupervised learning). Разумеется, в этом случае требуется некоторая информация о структуре полезного входного сигнала

(например, знание типа и параметров используемой модуляции). Очевидно, что адаптация по косвенным данным является более сложной вычислительной задачей, чем адаптация с использованием образцового сигнала. Некоторые сведения о ней можно найти в [4, 5].

Может показаться, что алгоритмы с использованием образцового сигнала не имеют практического значения, поскольку выходной сигнал должен быть заранее известен. Однако есть целый ряд прикладных задач, при решении которых образцовый сигнал оказывается доступен. Следует отметить, что в ряде случаев при этом полезным сигналом является не выходной сигнал фильтра, а *сигнал ошибки*, т.е. разность между образцовым сигналом и выходным сигналом адаптивного фильтра.

В качестве фильтра в структуре, показанной на рис. 1, чаще всего используется нерекурсивный цифровой фильтр. Главное его достоинство по сравнению с рекурсивным фильтром – устойчивость при любых значениях коэффициентов. Однако следует помнить, что алгоритм адаптации в любом случае вносит в систему обратную связь, вследствие чего адаптивная система в целом может стать неустойчивой.

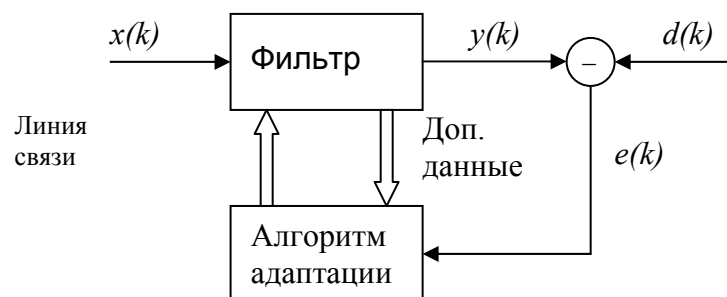


Рис. 1. Общая структура адаптивного фильтра

Существуют адаптивные алгоритмы и для рекурсивных фильтров, однако при их разработке возникают серьезные проблемы, прежде всего связанные с устойчивостью, поэтому такие фильтры не получили широкого распространения. Некоторые теоретические сведения на эту тему содержатся в [2]. Еще один класс адаптивных систем – *нейронные сети* (neural networks), которые в определенной степени моделируют работу нервной системы живых организмов.

Далее будут рассмотрены два адаптивных алгоритма с использованием образцового сигнала, часто применяемых на практике в различных системах обработки информации. Для упрощения математических выкладок предположим, что сигналы и фильтры являются вещественными. Однако результирующие формулы легко обобщаются на случай комплексных сигналов и фильтров.

2. ОПТИМАЛЬНЫЙ ФИЛЬТР ВИНЕРА

Прежде чем рассматривать собственно алгоритмы адаптации, необходимо определить оптимальные параметры фильтра. Подход к задаче оптимальной фильтрации может быть как статистическим, так и детерминированным. Сначала рассмотрим статистический вариант.

Пусть входной дискретный случайный сигнал $x(k)$ обрабатывается нерекурсивным дискретным фильтром порядка N , коэффициенты которого могут быть представлены вектор-столбцом $w = [w_0, w_1, \dots, w_N]^T$. Выходной сигнал фильтра

$$y(k) = u^T(k)w, \quad (2.1)$$

где $u(k) = [x(k), x(k-1), \dots, x(k-N)]^T$ - вектор-столбец содержимого линии задержки фильтра на k -м шаге.

Кроме того, имеется образцовый (также случайный) сигнал $d(k)$. Ошибка воспроизведения образцового сигнала

$$e(k) = d(k) - y(k) = d(k) - u^T(k)w. \quad (2.2)$$

Необходимо найти такие коэффициенты фильтра w , которые обеспечивают максимальную близость выходного сигнала фильтра к образцовому, то есть минимизируют ошибку $e(k)$. Поскольку $e(k)$ также является случайным процессом, в качестве меры ее величины разумно принять средний квадрат. Таким образом, оптимизируемый функционал выглядит так:

$$J(w) = \overline{e^2(k)} \rightarrow \min.$$

Квадрат ошибки

$$e^2(k) = (d(k) - u^T(k)w)^2 = \\ = d^2(k) - 2d(k)u^T(k)w + w^T u(k)u^T(k)w.$$

Статистически усредняя это выражение, получаем следующее:

$$J(w) = \overline{e^2(k)} = \overline{d^2(k)} - 2\overline{d(k)u^T(k)w} + w^T \overline{u(k)u^T(k)}w. \quad (0.1)$$

Входящие в полученную формулу усредненные величины имеют следующий смысл:

$\overline{d^2(k)} = \sigma_d^2$ – средний квадрат образцового сигнала;

$\overline{d(k)u^T(k)} = r^T$ – транспонированный вектор-столбец r взаимных корреляций между k -м отсчетом образцового сигнала и содержимым линии задержки фильтра. Если рассматриваемые случайные процессы $x(t)$ и $d(k)$ являются совместно стационарными, вектор взаимных корреляций не зависит от номера шага k ;

$\overline{u(k)u^T(k)} = R$ – корреляционная матрица сигнала, имеющая размер $(N+1) \cdot (N+1)$. Для стационарного случайного процесса корреляционная матрица имеет вид матрицы Теплица, т. е. на ее диагоналях стоят одинаковые величины:

$$R = \begin{bmatrix} R_x(0) & R_x(1) & R_x(2) & \dots & R_x(N) \\ R_x(1) & R_x(0) & R_x(1) & \dots & R_x(N-1) \\ R_x(2) & R_x(1) & R_x(0) & \dots & R_x(N-2) \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ R_x(N) & R_x(N-1) & R_x(N-2) & \dots & R_x(0) \end{bmatrix},$$

здесь $R_x(m) = \overline{x(k)x(k-m)}$ – корреляционная функция (КФ) случайного процесса $\{x(k)\}$.

С учетом введенных обозначений **Ошибка! Источник ссылки не найден.** принимает следующий вид:

$$J(w) = \sigma_d^2 - 2p^T w + w^T R w. \quad (0.2)$$

Данное выражение представляет собой квадратичную форму относительно w , и потому при невырожденной матрице R имеет единственный минимум, для нахождения которого необходимо приравнять нулю вектор градиента:

$$\text{grad} J(w) = -2p + 2Rw = 0.$$

Отсюда получаем искомое решение для оптимальных коэффициентов фильтра:

$$w = R^{-1} p. \quad (0.3)$$

Такой фильтр называется *фильтром Винера*. Подстановка **Ошибка! Источник ссылки не найден.** в **Ошибка! Источник ссылки не найден.** дает минимально достижимую дисперсию сигнала ошибки:

$$\overline{e^2(k)}_{\min} = \sigma_d^2 - p^T R^{-1} p. \quad (0.4)$$

Несложно также показать, что $\overline{e(k)y(k)} = 0$ и $\overline{e(k)x(k)} = 0$, т.е. что сигнал ошибки для фильтра Винера некоррелирован со входным и выходным сигналами фильтра.

В качестве примера рассчитаем с помощью MATLAB оптимальный фильтр для коррекции искажений, вносимых в передаваемый сигнал $x_0(k)$ каналом связи, имеющим четырехэлементную импульсную характеристику следующего вида: $h = [-2, -4, 6, 3]$. Отсчеты передаваемого сигнала будем считать дискретным белым шумом с нулевым средним значением и единичной дисперсией. В этом случае корреляционная функция сигнала, прошедшего через канал связи, будет совпадать с корреляционной функцией импульсной характеристики канала: $r = [65, 2, -24, -6]$.

Корреляционная матрица входного сигнала строится как матрица Теплица на основе данной корреляционной функции:

$$R = \begin{bmatrix} 65 & 2 & -24 & -6 & 0 & \dots & 0 \\ 2 & 65 & 2 & -24 & -6 & \dots & 0 \\ -24 & 2 & 65 & 2 & -24 & \dots & 0 \\ -6 & -24 & 2 & 65 & 2 & \dots & 0 \\ 0 & -6 & -24 & 2 & 65 & \dots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & 0 & 0 & \dots & 65 \end{bmatrix}.$$

Восстановление переданного сигнала неизбежно требует внесения некоторой временной задержки, поэтому образцовый сигнал представляет собой задержанную копию переданного: $d(k) = x_0(k - \Delta k)$. В линии задержки фильтра на k -м шаге находятся отсчеты искаженного сигнала с номерами $k, k-1, k-2, \dots, k-N$, где N – порядок фильтра. Каждый из этих отсчетов представляет собой линейную комбинацию отсчетов переданного сигнала:

$$x(k) = \sum_{n=-\infty}^{\infty} x_0(n)h(k-n). \quad (0.5)$$

Поскольку исходный сигнал считается белым шумом, то при вычислении m -го элемента вектора p результат усреднения будет отличен от нуля только для одного слагаемого **Ошибка! Источник ссылки не найден.**:

$$p_m = \overline{x(k-m)d(k)} = \overline{\sum_{n=-\infty}^{\infty} x_0(n)h(k-m-n)x_0(k-\Delta k)} = h(\Delta k - m).$$

Таким образом, вектор p содержит *перевернутую* импульсную характеристику канала, при необходимости обрезанную или дополненную нулями с одной или двух сторон.

Ниже приводится код MATLAB-программы, реализующей расчет оптимального фильтра:

```

clear all;
h = [-2 -4 6 3]; % импульсная характеристика канала связи
N = 32; % порядок рассчитываемого фильтра
r = xcorr(h, N);
r = r(N+1:end); % односторонняя КФ импульсной характеристики канала
R = toeplitz(r); % корреляционная матрица искаженного сигнала
p = zeros(N+1, 1);
k0 = round((N-length(h)) / 2) ;
p(k0:k0 + length(h)-1) = fliplr(h); % вектор взаимных корреляций
w = R\p; % коэффициенты оптимального фильтра
subplot(1,2,1)
impz(w) % график импульсной характеристики рассчитанного фильтра
subplot(1,2,2)
impz(conv(h, w) ) % график импульсной характеристики
% скорректированного канала

```

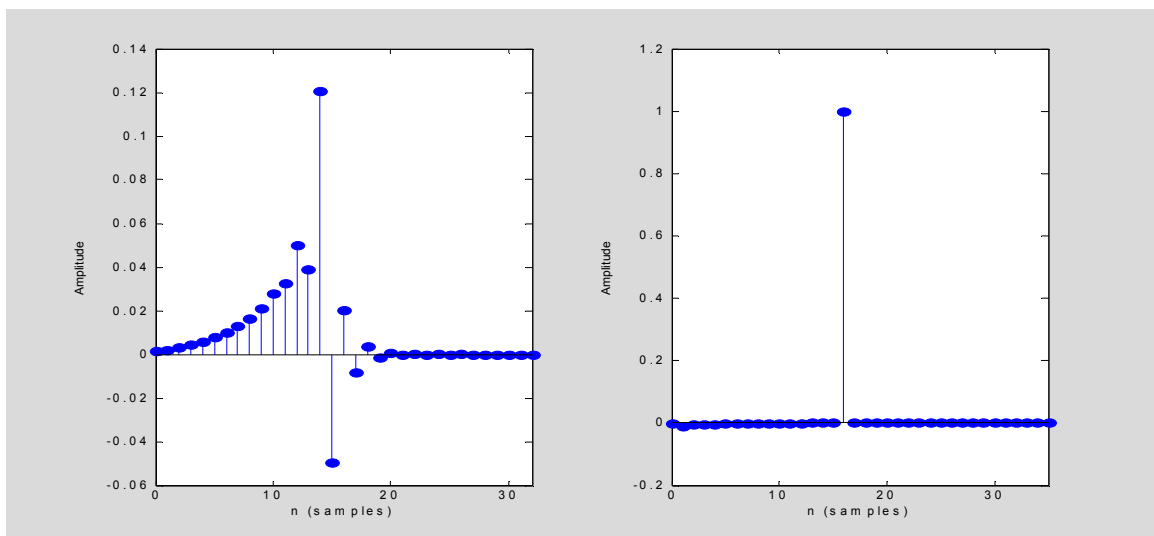


Рис. 2. Импульсные характеристики корректирующего фильтра и скорректированного канала связи

Код составлен так, чтобы можно было варьировать вид импульсной характеристики канала и порядок рассчитываемого фильтра.

На рис. 2 приведены графики импульсной характеристики корректирующего фильтра и сквозной импульсной характеристики скорректированного ка-

нала связи. Видно, что сквозная характеристика близка к единичному импульсу, что говорит о хорошем качестве коррекции. Столь качественная коррекция оказалась возможна потому, что частотная характеристика канала не имеет глубоких провалов.

3. АЛГОРИТМ LMS

Один из наиболее распространенных адаптивных алгоритмов основан на поиске минимума целевой функции **Ошибка! Источник ссылки не найден.** методом наискорейшего спуска. При использовании данного способа оптимизации вектор коэффициентов фильтра $w(k)$ должен рекурсивно обновляться следующим образом:

$$w(k+1) = w(k) - \frac{\mu}{2} \text{grad} J(w(k)) = w(k) + \mu p - \mu R w(k), \quad (3.1)$$

где μ – положительный коэффициент, называемый размером шага. Подробный анализ сходимости данного процесса приведен, например, в [2]. Показано, что алгоритм сходится, если $0 < \mu < 2/\lambda_{\max}$, где λ_{\max} – максимальное собственное число корреляционной матрицы R . Скорость сходимости при этом зависит от разброса собственных чисел корреляционной матрицы R – чем меньше отношение $\lambda_{\max}/\lambda_{\min}$, тем быстрее сходится итерационный процесс.

Однако для расчета градиента необходимо знать значения матрицы R и вектора p . На практике могут быть доступны лишь *оценки* этих значений, получаемые по входным данным. Такими простейшими оценками являются *мгновенные* значения корреляционной матрицы и вектора взаимных корреляций, получаемые без какого-либо усреднения:

$$\begin{aligned} R(k) &= u(k)u^T(k), \\ p(k) &= d(k)u(k). \end{aligned}$$

При использовании данных оценок формула (3.1) принимает следующий вид:

$$\begin{aligned}
 w(k+1) &= w(k) + \mu d(k)u(k) - \mu u(k)u^T(k)w(k) = \\
 &= w(k) + \mu u(k)(d(k) - u^T(k)w(k)).
 \end{aligned}
 \tag{3.2}$$

Выражение, стоящее в скобках, согласно, представляет собой разность между образцовым сигналом и выходным сигналом фильтра на k -м шаге, т.е. ошибку фильтрации $e(k)$. С учетом этого выражение для рекурсивного обновления коэффициентов фильтра оказывается очень простым:

$$w(k+1) = w(k) + \mu e(k)u(k). \tag{3.3}$$

Алгоритм адаптивной фильтрации, основанный на формуле(3.3), получил название LMS (Least Mean Square – метод наименьших квадратов). Можно получить ту же формулу и несколько иным образом: используя вместо градиента *статистически усредненного* квадрата ошибки $\overline{e^2(k)}$ градиент его *мгновенного значения* $e^2(k)$.

Анализ сходимости алгоритма LMS показывает [4], что верхняя граница для размера шага μ в данном случае является меньшей, чем при использовании истинных значений градиента. Эта граница μ_{\max} примерно равна

$$\mu_{\max} \approx \frac{2}{\sum_k \lambda_k} = \frac{2}{\text{trace}(\mathbf{R})} = \frac{2}{(N+1)\sigma_x^2}, \tag{3.4}$$

где λ_k – собственные числа корреляционной матрицы \mathbf{R} , а σ_x^2 – средний квадрат входного сигнала фильтра.

Основным достоинством алгоритма LMS является предельная вычислительная простота – для подстройки коэффициентов фильтра на каждом шаге нужно выполнить $N + 1$ операций умножения и сложения. Следствием этого являются медленная сходимость и повышенная (по сравнению с минимально достижимым значением **Ошибка! Источник ссылки не найден.**) дисперсия ошибки в установившемся режиме – коэффициенты фильтра всегда флуктуируют вокруг оптимальных значений **Ошибка! Источник ссылки не найден.**, что и

увеличивает уровень выходного шума. Ниже приводится код MATLAB-программы, реализующей LMS– алгоритм (рис. 3):

```
clear all;
NoOfData = 8000 ;           % Размерность входной последовательности
Order = 32 ;               % Порядок фильтра
Mu = 0.01 ;               %
x = randn(NoOfData, 1) ;   % Инициализация входной последовательности
h = rand(Order, 1) ;      % Инициализация ИХ канала связи
d = filter(h, 1, x) ;     %
w = zeros(Order,1) ;      % Инициализация ИХ адаптивного фильтра
% LMS алгоритм
for n = Order : NoOfData
    D = x(n:-1:n-Order+1) ;
    d_hat(n) = w'*D ;
    e(n) = d(n) - d_hat(n) ;
    w = w + Mu*e(n)*D ;
    w_err(n) = norm(h - w) ;
end ;
subplot (1,2,1); plot(20*log10(abs(e))) ;
subplot(1,2,2); semilogy(w_err) ;
```

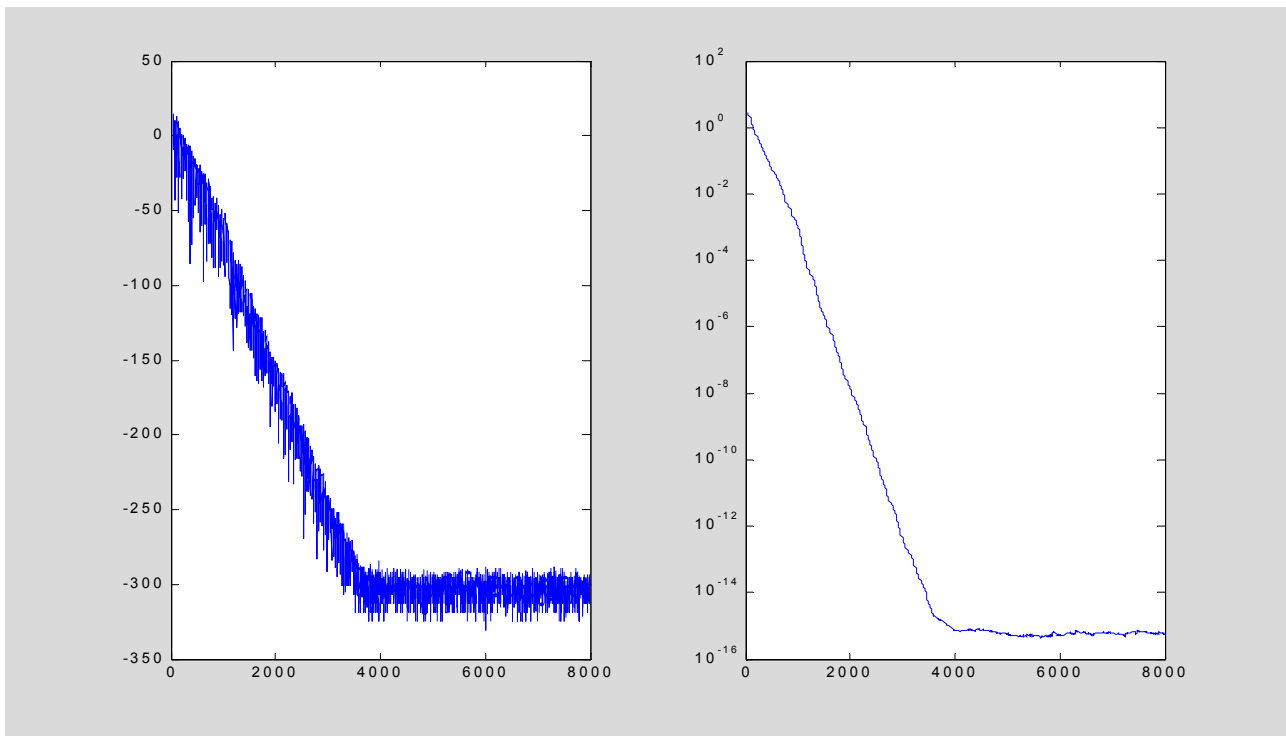


Рис. 3. Графики ошибки $e(n)$ и ошибки оценки корректируемой импульсной характеристики

Существует большое число модификаций алгоритма LMS [3], направленных на ускорение сходимости либо на уменьшение числа арифметических операций. Ускорение сходимости может быть достигнуто за счет улучшения используемой оценки градиента, а также за счет преобразования входного сигнала с целью сделать его отсчеты некоррелированными. Уменьшение вычислительной сложности может быть достигнуто, в частности, за счет использования в (3.3) не самих сигнала ошибки и содержимого линии задержки фильтра, а лишь их *знаков*. Это позволяет полностью избавиться от операций умножения при обновлении коэффициентов фильтра. В целом следует отметить, что требования ускорения сходимости и сокращения вычислительных затрат являются противоречивыми.

4. АДАПТИВНАЯ ФИЛЬТРАЦИЯ НЕСТАЦИОНАРНОГО СИГНАЛА

При рассмотрении статистической задачи оптимизации входной сигнал считался *случайным процессом* и минимизировался *средний квадрат*

ошибки воспроизведения образцового сигнала. В случае приема нестационарного сигнала статистическое усреднение может быть невозможно и требуется вести обработку по одной или нескольким реализациям процесса.

Пусть, как и раньше, обработке подвергается последовательность, состоящая из K отсчетов $x(k)$, коэффициенты нерекурсивного фильтра образуют вектор-столбец w , а отсчеты образцового сигнала равны $d(k)$. Выходной сигнал фильтра определяется формулой **Ошибка! Источник ссылки не найден.**, а ошибка воспроизведения образцового сигнала – формулой. Теперь оптимизационная задача формулируется так: нужно отыскать такие коэффициенты фильтра w , чтобы норма ошибки воспроизведения образцового сигнала была минимальной:

$$J(w) = \sum_{k=0}^{K-1} |e(k)|^2 \rightarrow \min. \quad (4.1)$$

Для решения задачи необходимо привести формулы и к матричной форме вдоль координаты k , получив формулы для вектор-столбцов выходного сигнала y и для ошибки воспроизведения входного сигнала e :

$$\begin{aligned} y &= U^T w, \\ e &= d - U^T w. \end{aligned} \quad (4.2)$$

Здесь d – вектор-столбец отсчетов образцового сигнала, а U – матрица, столбцы которой представляют собой содержимое линии задержки фильтра на разных тактах:

$$U = [u(0), u(1), \dots, u(K-1)].$$

Выражение (4.1) для нормы ошибки можно переписать в матричном виде следующим образом:

$$J(w) = e^T e \rightarrow \min. \quad (4.3)$$

Подставив (4.2) в (4.3), имеем

$$\begin{aligned}
 J(w) &= (d - U^T w)^T (d - U^T w) = \\
 &= d^T d - w^T U d - d^T U^T w + w U U^T w.
 \end{aligned}$$

Для нахождения минимума необходимо вычислить градиент данного функционала и приравнять его нулю:

$$\text{grad } J(w) = -2Ud + 2UU^T w = 0.$$

Отсюда легко получается искомое оптимальное решение:

$$w = (UU^T)^{-1} Ud. \quad (4.4)$$

В формуле (4.4) прослеживается близкое родство с формулой **Ошибка! Источник ссылки не найден.**, описывающей оптимальный фильтр Винера. Действительно, если учесть, что $(UU^T)^{-1}/K$ дает оценку корреляционной матрицы сигнала, полученную по одной реализации сигнала путем временного усреднения, и является аналогичной оценкой взаимных корреляций между образцовым сигналом и содержимым линии задержки фильтра, то формулы **Ошибка! Источник ссылки не найден.** и (4.4) совпадут.

5. АЛГОРИТМ RLS

В принципе, в процессе приема сигнала можно на каждом очередном шаге пересчитывать коэффициенты фильтра непосредственно по формуле (4.4), однако это связано с неоправданно большими вычислительными затратами. Действительно, размер матрицы U постоянно увеличивается и, кроме того, необходимо каждый раз заново вычислять обратную матрицу $(UU^T)^{-1}$.

Сократить вычислительные затраты можно, если учесть, что на каждом шаге к матрице U добавляется лишь один новый столбец, а к вектору d – один новый элемент. Это дает возможность организовать вычисления *рекурсивно*. Соответствующий алгоритм называется *рекурсивным методом наименьших квадратов* (Recursive Least Square – RLS).

Подробный вывод формул, описывающих алгоритм RLS, можно найти, например, в [2, 4]. При использовании алгоритма RLS производится рекурсивное обновление оценки обратной корреляционной матрицы $P = (UU^T)^{-1}$, а вывод формул основывается на следующем матричном тождестве:

$$(A + BCD)^{-1} = A^{-1} - A^{-1}B(C^{-1} + DA^{-1}B)^{-1}DA^{-1}, \quad (5.1)$$

где A и C – квадратные невырожденные матрицы (необязательно одинаковых размеров), а B и D – матрицы совместимых размеров.

Применение формулы (5.1) для рекурсивного обновления обратной корреляционной матрицы P в сочетании с исходной формулой (4.4) для коэффициентов оптимального фильтра обуславливает следующую последовательность шагов адаптивного алгоритма RLS:

1. При поступлении новых входных данных $u(k)$ производится фильтрация сигнала с использованием текущих коэффициентов фильтра $w(k-1)$ и вычисление величины ошибки воспроизведения образцового сигнала:

$$\begin{aligned} y(k) &= u^T(k)w(k-1), \\ e(k) &= d(k) - y(k). \end{aligned}$$

2. Рассчитывается вектор-столбец *коэффициентов усиления* (следует отметить, что знаменатель дроби в следующих двух формулах является скаляром, а не матрицей):

$$K(k) = \frac{P(k-1)u(k)}{1 + u^T(k)P(k-1)u(k)}. \quad (5.2)$$

3. Производится обновление оценки обратной корреляционной матрицы сигнала:

$$P(k) = P(k-1) - K(k)u^T(k)P(k-1). \quad (5.3)$$

4. Наконец, производится обновление коэффициентов фильтра:

$$w(k) = w(k-1) + K(k)e(k).$$

Начальное значение вектора w обычно принимается нулевым, а в качестве исходной оценки матрицы P используется диагональная матрица вида CI/σ_x^2 , где $C \gg 1$ (в [2] рекомендуется $C \geq 100$).

В формулах (5.1) и (5.3) значениям ошибки на всех временных тактах придается одинаковый вес. В результате, если статистические свойства входного сигнала со временем изменяются, это приводит к ухудшению качества фильтрации. Чтобы фильтр отслеживал нестационарный входной сигнал, можно применить в (5.1) экспоненциальное забывание, при котором вес прошлых значений сигнала ошибки экспоненциально уменьшается:

$$J(w) = \sum_{k=0}^{K-1} \lambda^{K-1-k} |e(k)|^2, 0 < \lambda \leq 1.$$

При использовании экспоненциального забывания формулы (5.2) и (5.3) принимают следующий вид:

$$K(k) = \frac{P(k-1)u(k)}{\lambda + u^T(k)P(k-1)u(k)},$$

$$P(k) = \frac{1}{\lambda} (P(k-1) - K(k)u^T(k)P(k-1)).$$

Главным достоинством алгоритма RLS является быстрая сходимость. Однако достигается это за счет значительного усложнения (по сравнению с алгоритмом LMS) вычислений. Согласно [2], при оптимальной организации вычислений для обновления коэффициентов фильтра на каждом такте требуется $(2,5N^2 + 4N)$ пар операций “умножение-сложение”.

6. АЛГОРИТМ КАЛМАНА

Назначение фильтра Калмана – минимизировать дисперсию оценки векторного случайного процесса $x(k)$, изменяющегося во времени следующим образом:

$$x(k+1) = \Psi(k)x(k) + v(k),$$

где $\Psi(k)$ — матрица перехода, $v(k)$ — случайный вектор (*шум процесса*), имеющий нормальное распределение с корреляционной матрицей $Q_p(k)$.

Для наблюдения доступен линейно преобразованный процесс $y(k)$, к которому добавляется шум наблюдения:

$$y(k) = H(k)x(k) + w(k),$$

где $H(k)$ — матрица наблюдения, $w(k)$ — шум наблюдения, представляющий собой случайный вектор, имеющий нормальное распределение с корреляционной матрицей $Q_M(k)$.

Поиск алгоритма для рекурсивного обновления оценки процесса $x(k)$ обуславливает следующую последовательность формул:

- $y(k) = C(k)\Psi(k)x(k-1)$ — прогнозируемое значение наблюдаемого сигнала;
- $e(k) = y(k) - \hat{y}(k)$ — ошибка между прогнозируемым и реально наблюдаемым значениями;
- $K(k) = P(k-1)C^T(k) \cdot [C(k)P(k-1)C^T(k) + Q_M(k)]^{-1}$ — калмановский коэффициент усиления;
- $\hat{x}(k) = \Psi(k)\hat{x}(k-1) + K(k)e(k)$ — обновление оценки процесса;
- $P(k) = \Psi(k)[P(k-1) - K(k)C(k)P(k-1)]\Psi^T(k) + Q_M(k)$ — обновление оценки корреляционной матрицы ошибок фильтрации.

При использовании фильтра Калмана для решения задачи адаптивной фильтрации отслеживаемым параметром является *вектор коэффициентов оптимального фильтра* w . Предполагается, что детерминированных изменений коэффициентов не происходит, поэтому матрица перехода Ψ является единичной: $\Psi(k) = I$. В качестве матрицы наблюдения выступает вектор содержимого линии задержки фильтра $u(k)$. Таким образом, выходной сигнал фильтра представляет собой прогнозируемое значение наблюдаемого сигнала, а в качестве самого наблюдаемого сигнала выступает образцовый сигнал адаптивного

фильтра $d(k)$. Шум наблюдения в данном случае является ошибкой воспроизведения образцового сигнала, а матрица Q_M превращается в скалярный параметр – средний квадрат сигнала ошибки. Как отмечается в [4], величина этого параметра слабо влияет на поведение алгоритма. Рекомендуемые значения $(0,001\dots 0,01)\sigma_d^2$.

Если фильтруется стационарный случайный процесс, коэффициенты оптимального фильтра являются постоянными и можно принять $Q_p = 0$. Чтобы фильтр отслеживал медленные изменения статистики входного сигнала, в качестве Q_p может использоваться диагональная матрица.

В результате приведенные выше формулы принимают следующий вид:

- $y(k) = u^T(k)w(k-1)$ – выходной сигнал фильтра (прогнозируемое значение образцового сигнала);

- $e(k) = d(k) - y(k)$ – ошибка фильтрации;

- $K(k) = \frac{P(k-1)u(k)}{u^T(k)P(k-1)u(k) + Q_M}$ – калмановский коэффициент усиления;

- $w(k) = w(k-1) + K(k)e(k)$ – обновление оценки коэффициентов фильтра;

- $P(k) = P(k-1) - K(k)u^T P(k-1) + Q_p$ – обновление оценки корреляционной матрицы ошибок оценивания.

Начальное значение вектора w обычно принимается нулевым, а в качестве исходной оценки матрицы P используется диагональная матрица вида CI .

Сравнивая формулы, описывающие алгоритмы RLS и Калмана, легко заметить их сходство. Вычислительная сложность и качественные параметры двух алгоритмов также оказываются весьма близкими. Разница заключается лишь в исходных посылах, использовавшихся при выводе формул, и в трактовке параметров алгоритмов. В некоторых источниках (см., например, [5]) алгоритмы RLS и Калмана применительно к адаптивной фильтрации отождествляются.

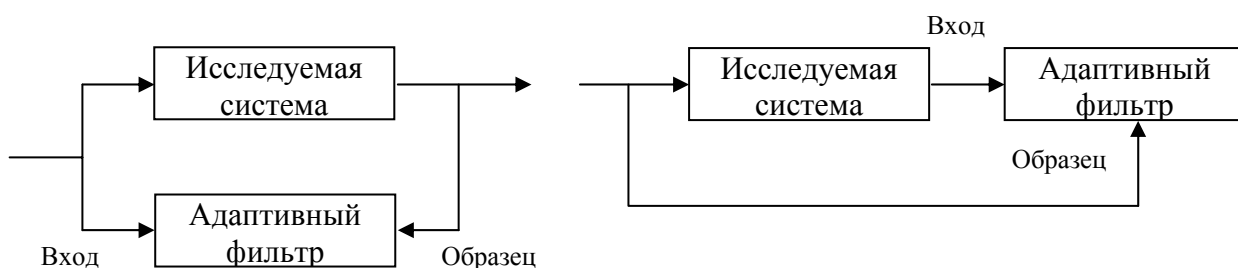
7. ПРИМЕНЕНИЕ АДАПТИВНЫХ ФИЛЬТРОВ

Адаптивные фильтры в настоящее время нашли применение во многих радиотехнических и телекоммуникационных системах. В данном разделе кратко рассматриваются лишь некоторые из возможных областей их использования.

7.1. Идентификация систем

Все способы использования адаптивных фильтров так или иначе сводятся к решению задачи *идентификации*, т.е. определения характеристик некоторой системы. Возможны два варианта идентификации – *прямая* и *обратная*. В первом случае адаптивный фильтр включается *параллельно* с исследуемой системой (рис.4, а). Входной сигнал является общим для исследуемой системы и адаптивного фильтра, а выходной сигнал системы служит для адаптивного фильтра образцовым сигналом. В процессе адаптации временные и частотные характеристики фильтра будут стремиться к соответствующим характеристикам исследуемой системы.

При обратной идентификации адаптивный фильтр включается *последовательно* с исследуемой системой (рис. 4, б). Выходной сигнал системы поступает на вход адаптивного фильтра, а входной сигнал системы является образцом для адаптивно-



го фильтра. Таким образом, фильтр компенсирует влияние системы и восстанавливает исходный сигнал, устранив внесенные системой искажения.

а

б

Рис. 4. Идентификация систем с помощью адаптивного фильтра:

а – прямая, б – обратная

В MATLAB прямая идентификация нестационарной системы с помощью адаптивного фильтра производится в демонстрационной программе **Adaptkalman-demo** из пакета расширения Filter Design, а также в модели **Kalmnsce** из набора блоков DSP Blokset среды моделирования Simulink.

Теперь перейдем от обобщенных схем к рассмотрению более конкретных вариантов.

7.2. Подавление шума

Пусть необходимо обеспечить, например, пилота самолета системой речевой связи. При этом воспринимаемый микрофоном речевой сигнал неизбежно окажется сильно зашумленным звуками работающего двигателя и т.п. Избавиться от этих шумов нельзя, но можно получить *образец* шумового сигнала, установив второй микрофон в непосредственной близости от двигателя (или иного источника шумов). Разумеется, этот шум нельзя просто вычесть из речевого сигнала, поскольку до двух микрофонов шум *следует разными путями* и, следовательно, претерпевает разные *искажения* (рис. 5). Однако шумовые случайные процессы, воспринимаемые двумя микрофонами, будут *коррелированными*, так как они происходят из общего источника. В то же время очевидно, что шумовой сигнал не коррелирован с полезным речевым сигналом.

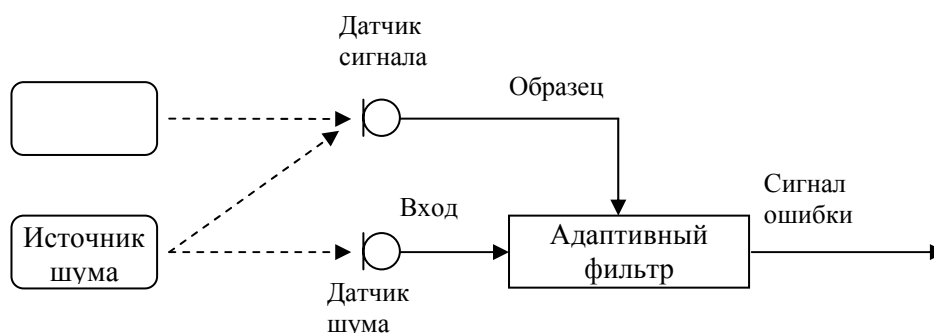


Рис. 5. Подавление шума с помощью адаптивного фильтра

С помощью адаптивного фильтра в данном случае решается задача прямой идентификации (см. рис. 4, а) преобразований шума на пути от шумового микрофона к сигнальному. Входным сигналом адаптивного фильтра является шумовой сигнал от дополнительного микрофона, а в качестве образцового сигнала используется сигнально-шумовая смесь, воспринимаемая основным микрофоном.

Адаптивный фильтр стремится преобразовать входной сигнал так, чтобы сделать его как можно ближе к образцовому. Поскольку со входным сигналом фильтра коррелирована лишь шумовая составляющая образцового сигнала, в установившемся режиме на выходе фильтра будет получена оценка шума, присутствующего в образцовом сигнале. Сигнал ошибки, рассчитываемый как разность между образцовым сигналом и выходным сигналом адаптивного фильтра, будет в этом случае представлять собой очищенный от шума речевой сигнал.

Для решения данной задачи в пакете Matlab можно использовать несколько демонстрационных программ, имеющих в составе MATLAB, – это программа **Adaptrlsdemo** из пакета расширения Filter Design, а также модели **Lmsdemo** и **Rlsdemo** из набора блоков DSP Blokset среды моделирования Simulink.

7.3. Выравнивание характеристик канала связи

При передаче по каналу связи информационный сигнал неизбежно претерпевает некоторые искажения. В системах цифровой связи эти искажения могут привести к возникновению ошибок при приеме цифровых данных. Для снижения вероятности ошибок необходимо компенсировать влияние канала связи, т.е. решить задачу обратной идентификации (см. рис. 4, б). В частотной области компенсация вносимых каналом искажений означает *выравнивание* (equalization) его частотной характеристики, поэтому фильтры, выполняющие такое выравнивание, получили название *эквалайзеров*.

При использовании адаптивного фильтра в качестве эквалайзера возникает проблема получения образцового сигнала. Эта проблема решается путем передачи специального настроечного сигнала перед началом передачи данных. В качестве такого настроечного сигнала обычно используется псевдослучайная последовательность символов. Алгоритм формирования этого сигнала известен приемной стороне, поэтому образцовый сигнал может быть сгенерирован автономно и использован для “обучения” адаптивного фильтра. Этот режим работы называется *режимом обучения* (training mode) (рис. 6).

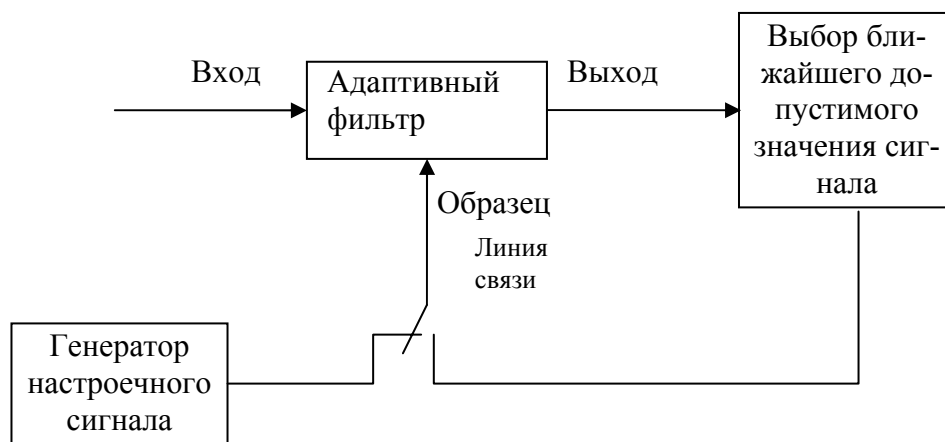


Рис. 6. Выравнивание канала связи с помощью адаптивного фильтра

После окончания настроечного сигнала начинается собственно передача данных. Приемник при этом переключается в другой режим, называемый *режимом оценивания* (decision-direct mode). В этом режиме для получения образцового сигнала используется тот факт, что множество возможных значений сигнала в системе цифровой связи является конечным. После приема очередного временного такта ищется ближайшее к принятому сигналу допустимое значение. Оно используется в качестве образцового сигнала, а разность между этим значением и принятым сигналом дает сигнал ошибки, используемый для адаптации.

Если после настройки эквалайзера, произведенной в режиме обучения, уровень шумов на выходе фильтра такой, что ближайшая допустимая точка в боль-

шинстве случаев оказывается правильной (т.е. если вероятность ошибки мала), адаптивный алгоритм сохраняет стабильность.

В некоторых, в частности многопользовательских, системах связи передавать настроенный сигнал не представляется возможным. В этом случае может использоваться (blind) адаптация по косвенным данным, некоторые алгоритмы которой приведены в [4, 5].

Адаптивное выравнивание канала связи можно осуществить при помощи демонстрационной модели **Lmsadeq** из набора блоков DSP Blokset среды моделирования Simulink.

7.4. Эхоподавление

Данная технология, так же как и выравнивание канала связи, широко используется в современных модемах. Скоростные модемы для телефонных линий связи работают *в дуплексном режиме*, т.е. передают и принимают данные одновременно, при этом для передачи и приема используется одна и та же полоса частот. Однако сигнал собственного передатчика в данном случае неизбежно просачивается в приемник, мешая работе последнего. Просачивающийся сигнал может распространяться разными путями, приобретая при этом неизвестные заранее искажения.

Подавить эхо-сигнал можно с помощью адаптивного фильтра. При этом решается задача прямой идентификации тракта распространения эха (рис. 7). На вход адаптивного фильтра поступает сигнал передатчика модема, а в качестве образцового сигнала используется принимаемый сигнал, содержащий эхо. Адаптивный фильтр формирует оценку эхо-сигнала, а сигнал ошибки представляет собой принимаемый сигнал, очищенный от эха.

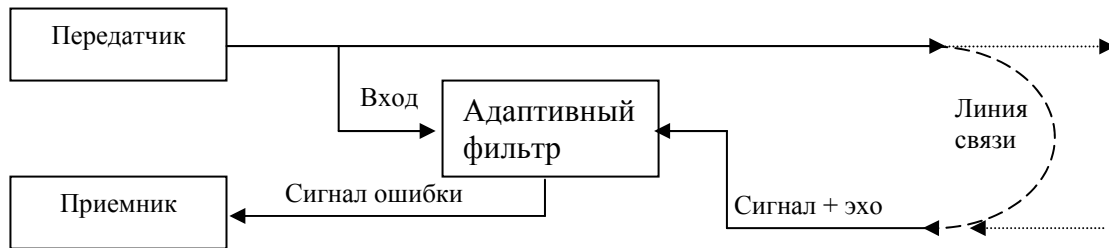


Рис. 7. Эхоподавление при помощи адаптивного фильтра

Для правильной работы системы эхоподавления необходимо, чтобы передаваемый и принимаемый сигналы были некоррелированы. Поэтому входные данные, поступающие в модем для передачи, прежде всего подвергаются *скремблированию* (scrambling), т.е. преобразуются в псевдослучайный битовый поток. При этом два взаимодействующих модема используют разные скремблеры, что и обеспечивает некоррелированность.

Эхоподавление, осуществляемое согласно рис. 7, используется во всех современных модемах.

8. ФУНКЦИИ АДАПТИВНОЙ ФИЛЬТРАЦИИ В ПАКЕТЕ MATLAB

В пакете расширения Filter Design, начиная с версии 2.1, входящей в поставку MATLAB 6.1, содержатся функции, реализующие следующие алгоритмы адаптивной фильтрации:

- **adaptlms** – алгоритм LMS в классическом виде;
- **adaptnlms** – нормированный алгоритм LMS (расчет шага μ на каждом такте осуществляется автоматически по формуле (3.4), исходя из энергии фрагмента сигнала, содержащегося в линии задержки фильтра);
- **adaptsd** – алгоритм LMS, для адаптации используется только знак данных, содержащихся в линии задержки фильтра (sign-data);
- **adaptse** – алгоритм LMS, для адаптации используется только знак сигнала ошибки (sign-error);

• **adaptss** – алгоритм LMS, для адаптации используются только знаки сигнала ошибки и данных, содержащихся в линии задержки фильтра (sign-sign);

• **adaptrls** – алгоритм RLS, в том числе с экспоненциальным забыванием данных;

• **adaptkalman** – алгоритм Калмана.

Все семь функций имеют одинаковый синтаксис вызова:

$[y, e, a] = \text{adaptlms}(x, d, s);$

$[y, e, a] = \text{adaptnlms}(x, d, s);$

$[y, e, a] = \text{adaptsd}(x, d, s);$

$[y, e, a] = \text{adaptse}(x, d, s);$

$[y, e, a] = \text{adaptss}(x, d, s);$

$[y, e, a] = \text{adaptrls}(x, d, s);$

$[y, e, a] = \text{adaptkalman}(x, d, s),$

где x – вектор отсчетов входного сигнала; d – вектор отсчетов образцового сигнала; s – структура с параметрами алгоритма и начальным состоянием фильтра. Три возвращаемых результата имеют следующий смысл: y – вектор отсчетов выходного сигнала; e – вектор отсчетов сигнала ошибки; s – результирующие значения параметров и конечное состояние фильтра.

Структура s хранит текущее состояние фильтра и содержит параметры, набор которых зависит от конкретного алгоритма. Назначение полей структуры и соответствие входных параметров *init*-функций полям возвращаемой структуры s приведены в таблице. Доступ к начальному и конечному состояниям фильтра позволяет организовать блочную или даже поотсчетную обработку исходных данных.

Назначение полей структуры адаптивного фильтра

Поле структуры	Параметр функции init	Назначение поля структуры и параметра	Имеет смысл
coeffs	w0	Вектор коэффициентов фильтра. По длине этого вектора определяется порядок фильтра	Для всех алгоритмов
states	zi	Вектор содержимого линии задержки фильтра	--/
iter	Только для чтения	Число выполненных итераций	--/
step	mu	Размер шага μ	Для всех разновидностей LMS
leakage	lf	Коэффициент утечки (в диапазоне 0...1, по умолчанию равен 1)	--/
offset	offset	Константа, добавляемая к знаменателю формулы для ограничения роста μ , при малом уровне сигнала	Для нормированного LMS
invcov	p0	Оценка обратной корреляционной матрицы входного сигнала	Для RLS
lambda	lambda	Коэффициент забывания (в диапазоне 0...1)	--/
alg	alg	Вариант RLS-алгоритма: 'direct' – стандартный, 'sqrt' - обладающий большей численной устойчивостью	--/
gain	Только для чтения	Рассчитанный на последней выполненной итерации вектор коэффициентов усиления	Для RLS и алгоритма Калмана
errcov	k0	Оценка корреляционной матрицы ошибок отслеживания вектора коэффициентов фильтра	Для алгоритма Калмана
measvar	qm	Дисперсия шума наблюдения, т.е. ожидаемая дисперсия сигнала ошибки	--/
procov	qp	Корреляционная матрица ожидаемых флуктуаций коэффициентов фильтра	--/

В качестве примера реализуем адаптивную коррекцию искажений, вносимых в сигнал тем же каналом связи, что использовался в примере расчета фильтра Винера. При этом сравним качество работы четырех версий алгоритма LMS – классического варианта и трех вариантов с использованием знаковых преобразований. Чтобы более наглядно показать качество компенсации, в виде входного сигнала используем четырехуровневый цифровой сигнал, с равной

вероятностью принимающий значения -3, -1, 1 и 3. Код MATLAB-программы приведен ниже:

```
clear all;
h = [-2 -4 6 3];
Nx = 4000;
levels = [-3 -1 1 3];
x = randsrc (Nx, 1, levels);
y = conv (x, h);
N = 32;
mu = 1/var(levels,1)/sum(h.^2)/(N+1);
s_lms = initlms (zeros(N+1,1), mu);
s_lms_se = initse (zeros(N+1,1), mu/10);
s_lms_sd = initsd (zeros(N+1,1), mu);
s_lms_ss = initss (zeros(N+1,1), mu);
y = y (round(N/2)+1:end);
[x,y] = eqtflength (x, y);
[x1_lms, e_lms, s_lms] = adaptlms (y(1:1000), x(1:1000), s_lms);
[x1_lms_se, e_lms_se, s_lms_se] = adaptse (y, x, s_lms_se);
[x1_lms_sd, e_lms_sd, s_lms_sd] = adaptsd (y, x, s_lms_sd);
[x1_lms_ss, e_lms_ss, s_lms_ss] = adaptss (y, x, s_lms_ss);
subplot (4, 2, 1), plot (e_lms), title ('LMS')
subplot (4, 2, 2), plot (x1_lms, '.')
subplot (4, 2, 3), plot (e_lms_se), title ('LMS- sign error')
subplot (4, 2, 4), plot (x1_lms_se, '.')
subplot (4, 2, 5), plot (e_lms_sd), title ('LMS- sign data')
subplot (4, 2, 6), plot (x1_lms_sd, '.')
subplot (4, 2, 7), plot (e_lms_ss), title ('LMS- sign sign')
subplot (4, 2, 8), plot (x1_lms_ss, '.')
```

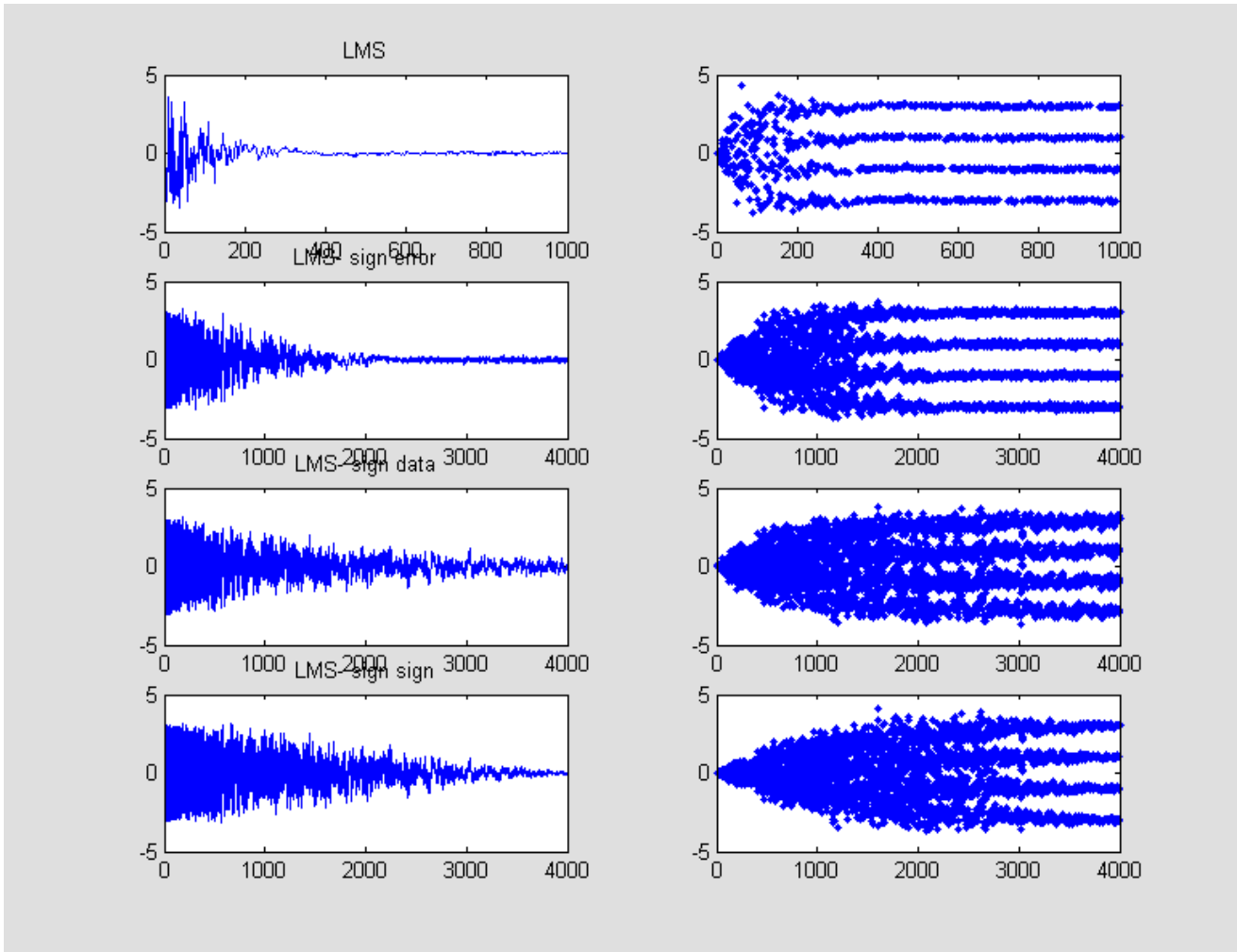


Рис. 8. Пример использования функций адаптивной фильтрации из пакета Filter Design

На рис. 8 показаны результаты работы программы. В левом столбце – зависимость сигнала ошибки от номера шага, в правом – график выходного сигнала фильтра. Видно, что использование знаковых преобразований замедляет сходимость LMS-алгоритма (пришлось даже увеличить число отображаемых итераций с 1000 до 4000). В наименьшей степени сходимость замедляется при знаковом преобразовании сигнала ошибки (sign-error), в наибольшей – при одновременном знаковом преобразовании сигнала ошибки и содержимого линии задержки фильтра (sign-data). Кроме того, предельное значение μ для знаковых вариантов сильно зависит от масштаба уровней образцового и входного сигналов. Так, в данном

примере пришлось в 10 раз уменьшить значение μ для варианта со знаковым преобразованием сигнала ошибки.

9. БЛОКИ АДАПТИВНОЙ ФИЛЬТРАЦИИ ИЗ НАБОРА DSP BLOKSET

В состав библиотеки Filtering/ Adaptive Filters набора блоков DSP Blokset среды моделирования Simulink входят три блока адаптивных фильтров:

- nLMS – алгоритм LMS, в том числе нормированный;
- RLS – алгоритм RLS, в том числе с экспоненциальным забыванием данных;
- Kalman – алгоритм Калмана.

Каждый из блоков по умолчанию имеет два входа (In – входной сигнал, Err – сигнал ошибки) и два выхода (Out – выходной сигнал, Taps – вектор коэффициентов фильтра). В отличие от функций пакета Filter Design вместо образцового сигнала на вход блока подается *сигнал ошибки*, который должен формироваться внешними цепями. Это позволяет использовать блоки более гибко. Параметры блоков в целом аналогичны полям структуры s в функциях пакета Filter Design, основные отличия заключаются в следующем:

- порядок фильтра задается явно – в виде числовой константы;
- для LMS не поддерживаются вариант с утечкой, а также варианты со знаковым преобразованием содержимого линии задержки фильтра (знаковое преобразование сигнала ошибки легко произвести во внешней цепи);
- начальные значения обратной корреляционной матрицы в алгоритме RLS и корреляционной матрицы сигнала ошибки в алгоритме Калмана принимаются диагональными и задаются путем указания скалярного параметра.

Среди параметров всех блоков имеется флажок Adapt input, при установке которого в блоке появляется третий вход Adapt. Этот вход позволяет включать и выключать адаптацию во время работы модели. При отключенной адаптации блок работает как фильтр с постоянными параметрами.

10. СОДЕРЖАНИЕ И ПОРЯДОК ВЫПОЛНЕНИЯ ЛАБОРАТОРНОЙ РАБОТЫ

10.1. Программное обеспечение лабораторной работы

Рассмотрим более детально решение задачи идентификации канала связи с помощью адаптивного фильтра, использующего алгоритм LMS, в пакете MatLab. В качестве исходного сигнала используется псевдослучайная последовательность, канал связи – КИХ-фильтр, сформированный с помощью блока Digital Filter Design. Входной сигнал "data" формируется блоком From WorkSpace по данным, находящимся в рабочей области пакета MatLab. В данном примере – это псевдослучайная последовательность длиной 31 отсчет. Источником шума в канале связи служит блок Random Source из набора DSP Blokset, позволяющий выбрать необходимые параметры случайной последовательности (рис. 9).

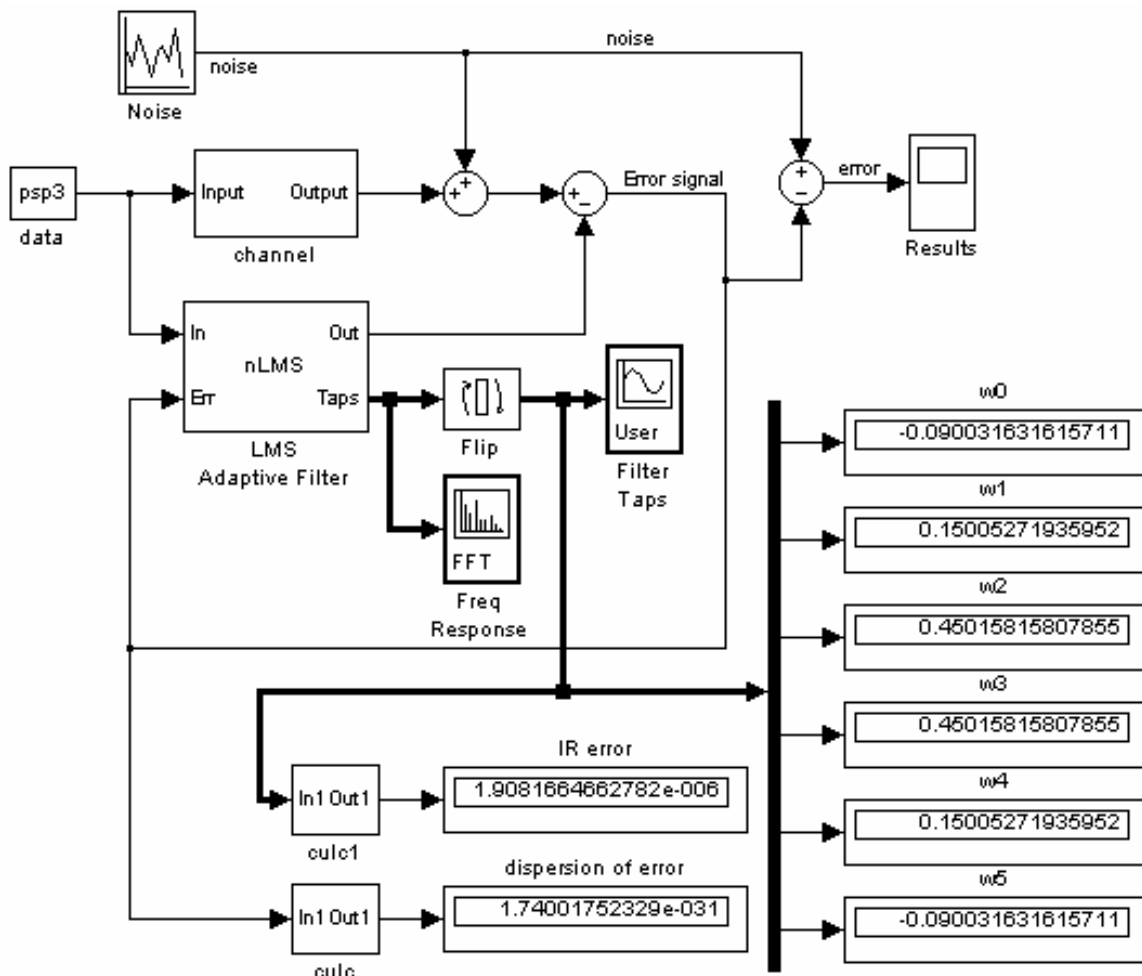


Рис. 9. Пример использования блока адаптивного фильтра из набора DSP Blokset

Скорость схождения LMS-алгоритма можно оценить по графику сигнала ошибки, снятого при помощи блока Score (рис.10), из которого видно, что сигнал ошибки практически не изменяется после 250 отсчетов.

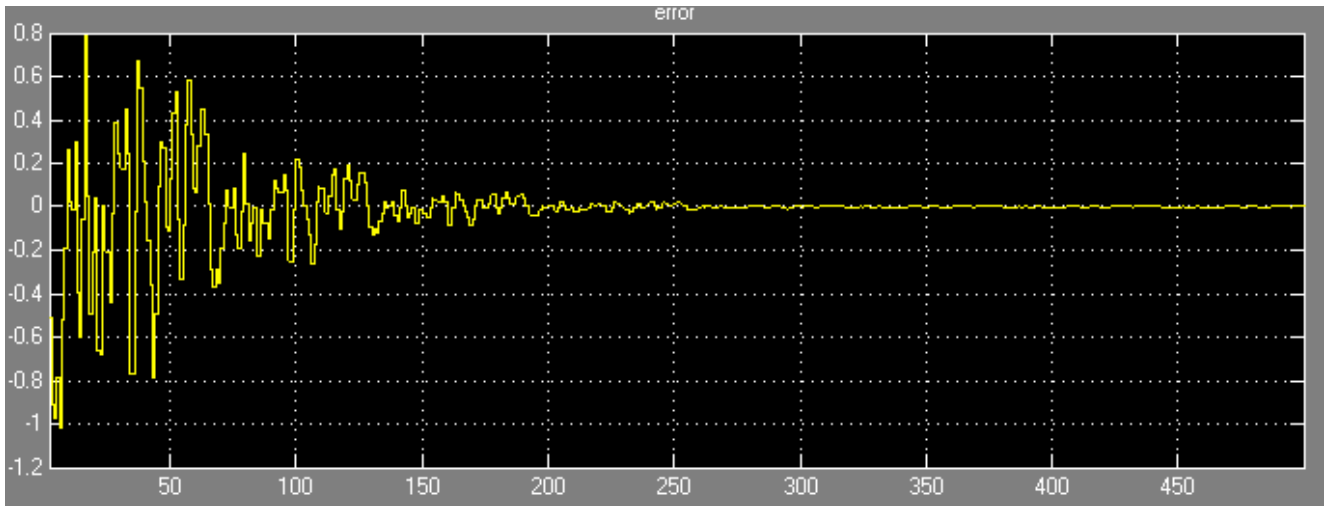


Рис.10. Сигнал ошибки

Блоки Display $w_0...w_5$ отображают коэффициенты импульсной характеристики адаптивного фильтра, блоки Spectrum score и Vector score (рис. 11) – соответственно частотную и импульсную характеристики адаптивного фильтра.

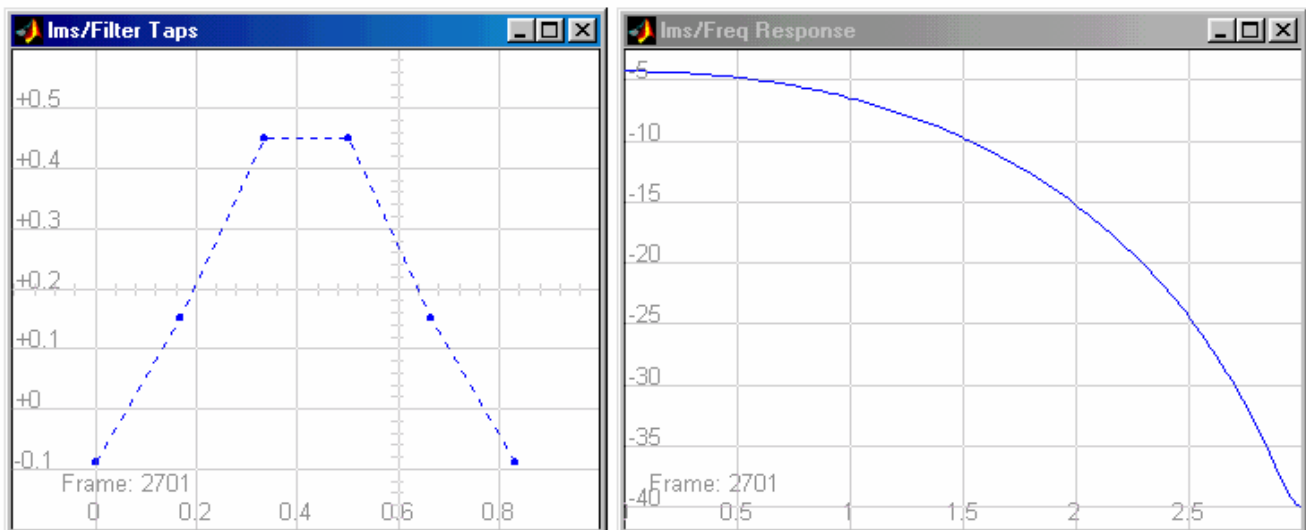


Рис.11. Импульсная и частотная характеристики адаптивного фильтра

Точность оценки импульсной характеристики канала связи можно определить по показаниям блока IR error, отображающего среднеквадратичную ошибку импульсных характеристик адаптивного фильтра и канала связи. Также качество схождения

димости алгоритма LMS можно оценить путем сравнения коэффициентов соответствующих импульсных характеристик (рис. 9, 12), применив любой другой критерий оценки качества.

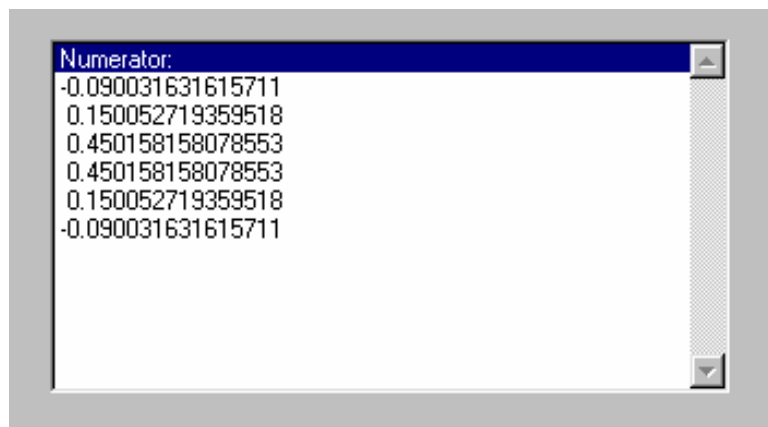


Рис.12. Окно Filter coefficients блока Digital Filter Design

В заключение следует отметить, что возможности функций пакета Filter Design и блоков набора DSP Blokset достаточны для использования в ознакомительных и учебных целях. Однако при попытке применить функции и блоки для профессионального моделирования реальных телекоммуникационных систем выявляется ряд недостатков и ограничений, присущих их реализации на момент написания пособия:

все функции адаптивной фильтрации полностью написаны на языке MATLAB, поэтому скорость их работы невелика. Это особенно относится к более сложным в вычислительном отношении алгоритмам Калмана и RLS;

как показывает синтаксис вызова функций, образцовый сигнал должен быть известен заранее, что делает невозможным применение данных функций в режиме оценивания (см. подраздел 7.3 и рис. 6);

функции и блоки на каждом временном такте реализуют сдвиг данных в линии задержки фильтра на один отсчет, что затрудняет применение функций в тех ситуациях, когда данные обновляются иным образом.

10.2. Содержание лабораторной работы

1. Изучение алгоритмов адаптивной обработки сигналов.
2. Изучение возможностей пакета MatLab для моделирования адаптивных систем.

10.3. Порядок выполнения лабораторной работы

1. Используя программу из разд. 2, изучить алгоритм синтеза оптимального фильтра Винера, оценить качество желаемой импульсной характеристики в зависимости от порядка адаптивного фильтра.
2. Используя программу из разд. 3, оценить скорость сходимости алгоритма LMS и дисперсию сигнала ошибки в установившемся режиме в зависимости от шага μ .
3. Собрать схему, представленную на рис. 9. Оценить качество идентификации канала связи в зависимости от типа входного сигнала, дисперсии шума и шага μ .

ЛИТЕРАТУРА

1. Уидроу Б., Стирнз С. Д. Адаптивная обработка сигналов. – М.: Радио и связь, 1989.
2. Адаптивные фильтры: Пер. с англ. / Под ред. К.Ф.Н. Коуэна и П.М. Гранта.– М.:Мир, 1988.
3. Glentis G.-O., Berberidis K., Theodoridis S. Efficient Least Squares Adaptive Algorithms for FIR Transversal Filtering// IEEE Signal Processing Magazine.- 1999. -V.16. -№.4. –P.13 -41.
4. Haykin S. Adaptive Filter Theory. 4-th edition.- Prentice Hall, 2002.
5. Проксис Дж. Цифровая связь: Пер. с англ./ Под ред. Д.Д. Кловского. – М.: Радио и связь, 2000.

Учебное издание

**Саломатин Сергей Борисович,
Бурак Андрей Иосифович**

**КОМПЬЮТЕРНОЕ МОДЕЛИРОВАНИЕ И ОБРАБОТКА
НЕСТАЦИОНАРНЫХ СИГНАЛОВ**

Методическое пособие к лабораторной работе по курсу
"Цифровая обработка сигналов"

для студентов специальностей "Радиотехника", "Радиоэлектронные системы"
всех форм обучения

Редактор Н.А. Бебель
Корректор Е.Н. Батурчик
Компьютерная верстка М.В. Шишло

Подписано в печать 20.07.2004.
Гарнитура «Таймс».
Уч.-изд. л. 1,2.

Формат 60x84 1/16.
Печать ризографическая.
Тираж 150 экз.

Бумага офсетная.
Усл. печ. л. 2,2.
Заказ 84.

Издатель и полиграфическое исполнение: Учреждение образования
«Белорусский государственный университет информатики и радиоэлектроники»
Лицензия на осуществление издательской деятельности №02330/0056964 от 01.04.2004.
Лицензия на осуществление полиграфической деятельности №02330/0133108 от 30.04.2004.
220013, Минск, П. Бровки, 6