

Быстрый алгоритм аффинных проекций: полная версия

В.И. Джиган

Федеральное государственное автономное образовательное учреждение высшего образования
«Национальный исследовательский университет «Московский институт электронной
техники», djigan@yandex.ru

Аннотация — В статье рассматривается быстрый алгоритм аффинных проекций для многоканальных адаптивных фильтров с комплексными весовыми коэффициентами. Приводится полная вычислительная процедура этого алгоритма, включающая линейное предсказание входных сигналов, используемое для быстрого, т.е. вычислительно эффективного, определения вектора нормализованных ошибок адаптивного фильтра. Эта процедура заимствована из стабилизированной версии быстрого рекурсивного алгоритма адаптивной фильтрации по критерию наименьших квадратов со скользящим окном. В рассматриваемом алгоритме аффинных проекций также используется процедура вычисления переменного шага сходимости, изменяемого по градиентному закону, и процедура вычисления выходного сигнала адаптивного фильтра. Даны оценки вычислительной сложности составляющих частей алгоритма. Приводятся примеры моделирования, подтверждающие его работоспособность и эффективность.

Ключевые слова — многоканальный адаптивный фильтр, быстрый алгоритм аффинных проекций, переменный шаг сходимости, вычисление выходного сигнала.

I. ВВЕДЕНИЕ

Сегодня существуют две основные группы алгоритмов адаптивной фильтрации сигналов, нашедших широкое применение в различных приложениях. Это вычислительно простые и вычислительно сложные алгоритмы. К простым относятся алгоритм по критерию наименьшего квадрата (Least Mean Square, LMS), нормализованный LMS-алгоритм (Normalized LMS, NLMS) и их разновидности, а к сложным – рекурсивные алгоритмы по критерию наименьших квадратов (Recursive Least Squares, RLS), включая быстрые (вычислительно эффективные) версии [1 – 4].

Простые алгоритмы сходятся медленно, длительность их переходных процессов и дополнительная среднеквадратичная ошибка в установившемся состоянии зависят от статистических характеристик входного сигнала адаптивного фильтра, чего нельзя сказать о сложных адаптивных алгоритмах. Однако из-за обычно ограниченных на практике программно-аппаратных ресурсов не всегда представляется возможным реализовывать сложные RLS-алгоритмы даже в их быстрой форме.

В то же время существует менее распространённый промежуточный класс алгоритмов адаптивной фильтрации, именуемый алгоритмами аффинных проекций (Affine Projection, AP) [5]. При обработке нестационарных сигналов AP-алгоритмы демонстрируют эффективность, близкую к эффективности RLS-алгоритмов, а благодаря наличию быстрой версии (Fast AP, FAP) [6, 7] они характеризуются вычислительной сложностью, близкой к сложности простых алгоритмов в случае использования в адаптивных фильтрах с большим числом весовых коэффициентов (сотни, тысячи). Такие адаптивные фильтры применяются при решении задач компенсации сигналов электрического и акустического эха [8, 9], выравнивания амплитудно-частотных характеристик каналов связи [10], активного шумоподавления [11], предсказания сигналов в усилителях мощности радиопередатчиков [12] и ряда других задач [13], в большинстве из которых наблюдаемым сигналом является выходной сигнал адаптивного фильтра.

Однако известная форма FAP-алгоритма [6, 7] не вычисляет выходной сигнал адаптивного фильтра. Она была разработана для решения задач эхо-компенсации, где наблюдаемым сигналом является лишь сигнал ошибки, т.е. подавленное эхо, который определяется в FAP-алгоритме без вычисления весовых коэффициентов и выходного сигнала адаптивного фильтра в явном виде. В настоящей работе показано, как можно вычислять выходной сигнал многоканального адаптивного фильтра, сохраняя при этом низкую вычислительную сложность FAP-алгоритма. Под сложностью понимается оценка числа арифметических операций, необходимых для выполнения одной итерации алгоритма, пропорциональная числу элементов используемых в вычислениях векторов.

Кроме того, в работе показано, как можно ускорить сходимость FAP-алгоритма за счет использования переменного шага сходимости, вычисляемого с помощью процедуры одномерного градиентного поиска.

Используя указанные выше дополнительные возможности, получена полная вычислительная процедура FAP-алгоритма для многоканального адаптивного фильтра с комплексными весовыми коэффициентами, которая приводится в данной работе.

Также приводятся результаты моделирования, демонстрирующие работоспособность и эффективность этого алгоритма.

II. АЛГОРИТМ АФФИННЫХ ПРОЕКЦИЙ

Существует несколько вариантов получения вычислительной процедуры АР-алгоритма. Однако, чтобы объяснить его эффективность при обработке нестационарных сигналов, где этот алгоритм получил наиболее широкое применение, лучше всего подходит следующий вариант.

Статистические характеристики нестационарных сигналов сильно меняются во времени. Это может приводить к большим изменениям весовых коэффициентов адаптивного фильтра от итерации к итерации и, как следствие, к возможно неустойчивой работе этого фильтра.

Целевая функция адаптивного фильтра обычно определяется как среднеквадратичная ошибка между его требуемым $d(k)$ и выходным $y(k)$ сигналами. Эта функция минимизируется в процессе работы фильтра, в результате чего находятся его весовые коэффициенты, соответствующие минимуму данной функции. Однако в АР-алгоритме целевую функцию можно определить и как норму разности векторов весовых коэффициентов на соседних итерациях алгоритма. Эта норма также минимизируется в процессе работы адаптивного фильтра. Чтобы при этом обеспечивалась и минимизация среднеквадратичной ошибки между сигналами $d(k)$ и $y(k)$, на целевую функцию накладывается линейное ограничение равенства нуль-вектору апостериорных ошибок на скользящем окне в L отсчетов обрабатываемых сигналов. Число L именуется размером проекции. Здесь k – номер итерации алгоритма, совпадающий с номером отсчета обрабатываемых сигналов.

Для адаптивного фильтра с комплексными весовыми коэффициентами указанное выше формализуется как:

$$\| \mathbf{h}_N(k) - \mathbf{h}_N(k-1) \|_2^2 \rightarrow \min ,$$

где $\| \bullet \|_2^2$ – евклидова норма вектора, при ограничении

$$\begin{aligned} \mathbf{e}_L^*(k) &= [e^*(k), e^*(k-1), \dots, e^*(k-L+1)]^T = \\ &= \mathbf{d}_L^*(k) - \mathbf{X}_{NL}^H(k) \mathbf{h}_N(k) = \mathbf{0}_L, \end{aligned}$$

где

$$\begin{aligned} \mathbf{d}_L(k) &= [d(k), d(k-1), \dots, d(k-L+1)]^T, \\ \mathbf{X}_{NL}(k) &= [\mathbf{x}_N(k), \mathbf{x}_N(k-1), \dots, \mathbf{x}_N(k-L+1)], \end{aligned}$$

а

$$\mathbf{x}_N(k) = [x(k), x(k-1), \dots, x(k-N_m+1)]^T$$

– вектор входных сигналов $x(k)$ фильтра.

Такая постановка задачи обеспечивает небольшие изменения значений весовых коэффициентов от итерации к итерации и минимизацию квадрата нормы

$$\mathbf{e}_L^T(k) \mathbf{e}_L^*(k) \rightarrow 0$$

скользящего вектора ошибок $\mathbf{e}_L(k)$.

В приведенных выше и следующих далее математических выражениях нижние индексы N , N_m и L обозначают число элементов векторов. Размер $N \times L$ прямоугольной матрицы $\mathbf{X}_{NL}(k)$ обозначается двумя индексами N и L , а размер $L \times L$ далее используемых квадратных матриц \mathbf{I}_L и $\mathbf{R}_L(k)$ обозначается одним нижним индексом L . Верхний индекс T обозначает операцию транспонирования вектора или матрицы, а верхний индекс H – операцию эрмитова сопряжения, т.е. транспонирования и комплексного сопряжения, обозначаемого символом $*$.

Решением этой задачи с использованием метода множителей Лагранжа является уравнение вычисления вектора весовых коэффициентов адаптивного фильтра

$$\mathbf{h}_N(k) = \mathbf{h}_N(k-1) + \mu \mathbf{X}_{NL}(k) \mathbf{e}_L(k),$$

т.е. АР-алгоритм, где

$$\begin{aligned} \mathbf{e}_L(k) &= [\mathbf{X}_{NL}^H(k) \mathbf{X}_{NL}(k) + \delta^2 \mathbf{I}_L]^{-1} \mathbf{a}_L^*(k), \\ \mathbf{e}_L(k) &= [\varepsilon(k), \varepsilon(k-1), \dots, \varepsilon(k-L+1)]^T, \\ \mathbf{a}_L(k) &= \mathbf{d}_L(k) - [\mathbf{h}_N^H(k-1) \mathbf{X}_{NL}(k)]^T, \end{aligned}$$

а δ^2 – параметр регуляризации при обращении матрицы.

Если адаптивный фильтр содержит M каналов (см. рис. 1), то вектор его весовых коэффициентов

$$\mathbf{h}_N(k) = [\mathbf{h}_{N_1}^T(k), \mathbf{h}_{N_2}^T(k), \dots, \mathbf{h}_{N_M}^T(k)]^T$$

составляется из векторов весовых коэффициентов каналов

$$\mathbf{h}_{N_m}(k) = [h_{1,m}(k), h_{2,m}(k), \dots, h_{N_m,m}(k)]^T,$$

а вектор сигналов

$$\mathbf{x}_N(k) = [\mathbf{x}_{N_1}^T(k), \mathbf{x}_{N_2}^T(k), \dots, \mathbf{x}_{N_M}^T(k)]^T$$

– из векторов сигналов каналов

$$\mathbf{x}_{N_m}(k) = [x_m(k), x_m(k-1), \dots, x_m(k-N_m+1)]^T.$$

Суммарное число весовых коэффициентов многоканального адаптивного фильтра определяется как $N = \sum_{m=1}^M N_m$. В зависимости от задачи, решаемой с

помощью адаптивного фильтра, число весовых коэффициентов в его каналах N_m в общем случае может быть не одинаковым.

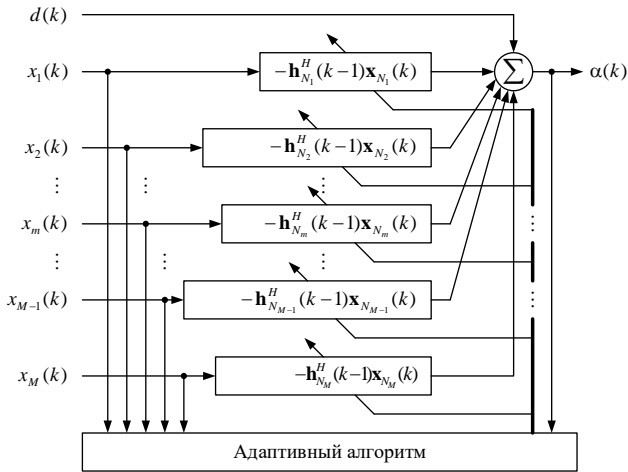


Рис. 1. Многоканальный адаптивный фильтр

III. МНОГОКАНАЛЬНЫЙ БЫСТРЫЙ АЛГОРИТМ АФФИННЫХ ПРОЕКЦИЙ

Быстрая версия одноканального AP-алгоритма [6, 7] основана на применении линейного предсказания для вычисления вектора нормализованных ошибок $\boldsymbol{\varepsilon}_L(k)$. Для этого могут быть использованы процедуры линейного предсказания из быстрых RLS-алгоритмов, в которых корреляционная матрица оценивается на прямоугольном скользящем окне отсчетов входных сигналов адаптивного фильтра [14, 15]. Однако в [6, 7] и ряде других работ не рассматривались особенности применения линейного предсказания в FAP-алгоритмах. Лишь в работе [16] впервые было показано, как в одноканальном FAP-алгоритме можно использовать процедуру линейного предсказания из быстрого RLS-алгоритма, именуемого быстрым трансверсальным фильтром (Fast Transversal Filter, FTF).

В настоящей работе представлена многоканальная версия FAP-алгоритма с комплексными весовыми коэффициентами, а также процедура линейного предсказания, используемая в быстром RLS-алгоритме, именуемом Fast Aposteriori Error Sequential Technique (FAEST).

При вычислении вектора нормализованных ошибок $\boldsymbol{\varepsilon}_L(k)$ в многоканальном AP-алгоритме применяется регуляризованная матрица

$$\begin{aligned} \mathbf{R}_L(k) &= \mathbf{X}_{NL}^H(k) \mathbf{X}_{NL}(k) + \delta^2 \mathbf{I}_L = \\ &= \sum_{m=1}^M \mathbf{X}_{N_m L}^H(k) \mathbf{X}_{N_m L}(k) + \delta^2 \mathbf{I}_L = \end{aligned}$$

$$\begin{aligned} &= \sum_{m=1}^M \sum_{i=0}^{N_m-1} \mathbf{x}_{L,m}(k-i) \mathbf{x}_{L,m}^T(k-i) + \delta^2 \mathbf{I}_L = \\ &= \sum_{m=1}^M \mathbf{R}_{L,m}(k) + \delta^2 \mathbf{I}_L. \end{aligned}$$

Используя данное уравнение и приемы, представленные [6, 7, 16], для многоканального адаптивного фильтра с комплексными весовыми коэффициентами разработан FAP-алгоритм (см. ниже) с переменным шагом сходимости $\mu(k)$ и вычислением выходного сигнала $y(k)$.

Инициализация: $\mathbf{x}_N(0) = \mathbf{0}_N$; $\hat{\mathbf{h}}_N(0) = \mathbf{0}_N$;
 $\mathbf{X}_{NL}(0) = \mathbf{0}_{NL}$; $\boldsymbol{\alpha}_L(0) = \mathbf{0}_L$; $\mathbf{p}_L(0) = \mathbf{0}_L$; $\boldsymbol{\varepsilon}_L(0) = \mathbf{0}_L$;
 $\mathbf{r}_L(k) = \mathbf{0}_L$; $\mathbf{s}_{L-1}(k) = \mathbf{0}_{L-1}$; $\mathbf{r}'_{L-1}(k) = \mathbf{0}_{L-1}$; $\mu(0) = \mu_0$

For $k=1, 2, \dots, K$

$$\begin{aligned} \mathbf{r}_L(k) &= \mathbf{r}_L(k-1) + \sum_{m=1}^M \left[x_m(k) \mathbf{x}_{L,m}^*(k-1) - \right. \\ &\quad \left. - x_m(k-N_m) \mathbf{x}_{L,m}^*(k-N_m-1) \right] \end{aligned}$$

$$\begin{aligned} \mathbf{r}'_{L-1}(k) &= \mathbf{r}'_{L-1}(k-1) + \sum_{m=1}^M \left[x'_m(k) \mathbf{x}_{L-1,m}^*(k-1) - \right. \\ &\quad \left. - x'_m(k-N_m) \mathbf{x}_{L-1,m}^*(k-N_m-1) \right] \end{aligned}$$

$$y(k) = \hat{\mathbf{h}}_N^H(k-1) \mathbf{x}'_N(k) + \bar{\mathbf{p}}_{L-1}^H(k-1) \mathbf{r}'_{L-1}(k)$$

$$\hat{\alpha}(k) = d(k) - \hat{\mathbf{h}}_N^H(k-1) \mathbf{x}_N(k)$$

$$\alpha(k) = \hat{\alpha}(k) - \bar{\mathbf{p}}_{L-1}^H(k-1) \mathbf{r}_{L-1}(k)$$

$$\boldsymbol{\alpha}_L(k) = \begin{bmatrix} \alpha(k) \\ [1 - \mu(k-1)] \bar{\boldsymbol{\alpha}}_{L-1}(k-1) \end{bmatrix}$$

$$\mu(k) = \mu(k-1) + 2\rho \operatorname{Re} \left[\alpha^*(k) \boldsymbol{\varepsilon}_L^H(k-1) \mathbf{r}_L(k) \right]$$

For $m=1, 2, \dots, M$

Вычисление: $E^f(k)$, $E^b(k)$, $\mathbf{h}_{L-1}^f(k)$, $\mathbf{h}_{L-1}^b(k)$

End for m

$$\boldsymbol{\varepsilon}_L(k) = \begin{bmatrix} 0 \\ \mathbf{s}_{L-1}(k-1) \end{bmatrix} + \frac{1}{E^f(k)} \begin{bmatrix} 1 \\ -\mathbf{h}_{L-1}^f(k) \end{bmatrix} \times$$

$$\times \begin{bmatrix} 1 \\ -\mathbf{h}_{L-1}^f(k) \end{bmatrix}^H \boldsymbol{\alpha}_L^*(k)$$

$$\begin{bmatrix} \mathbf{z}_{L-1}(k) \\ 0 \end{bmatrix} = \boldsymbol{\varepsilon}_L(k) - \frac{1}{E^b(k)} \begin{bmatrix} 1 \\ -\mathbf{h}_{L-1}^b(k) \end{bmatrix} \times$$

$$\times \begin{bmatrix} 1 \\ -\mathbf{h}_{L-1}^b(k) \end{bmatrix}^H \boldsymbol{\alpha}_L^*(k)$$

$$\mathbf{s}_{L-1}(k) = [1 - \mu(k)] \mathbf{z}_{L-1}(k)$$

$$\mathbf{p}_L(k) = \mu(k) \boldsymbol{\varepsilon}_L(k) + \begin{bmatrix} 0 \\ \bar{\mathbf{p}}_{L-1}(k-1) \end{bmatrix}$$

$$\hat{\mathbf{h}}_N(k) = \hat{\mathbf{h}}_N(k-1) + \mathbf{x}_N(k-L+1) p_L(k)$$

End for k

Обозначение $\bar{\mathbf{p}}_{L-1}(k-1)$ указывает на вектор, состоящий из первых $L-1$ элементов вектора $\mathbf{p}_L(k-1)$, а обозначение $p_L(k)$ – на L -й элемент вектора $\mathbf{p}_L(k)$. Аналогично, обозначение $\bar{\boldsymbol{\alpha}}_{L-1}(k-1)$ указывает на вектор, состоящий из первых $L-1$ элементов вектора $\boldsymbol{\alpha}_L(k-1)$.

IV. ПРОЦЕДУРА ЛИНЕЙНОГО ПРЕДСКАЗАНИЯ СИГНАЛОВ

Так как матрица $\mathbf{R}_L(k)$ состоит из суммы матриц, определяемых входными сигналами каждого из каналов многоканального адаптивного фильтра, то каждая из этих матриц определяется на собственном скользящем окне в N_m отсчетов. Для оценки обратного значения матрицы $\mathbf{R}_L^{-1}(k)$, используемой при вычислении вектора $\boldsymbol{\varepsilon}_L(k)$, на каждой итерации k требуется последовательно M раз применить одноканальное линейное предсказание к сигналам $x_m^*(k)$, $x_m^*(k-N_m)$, $x_m^*(k-L)$ и $x_m^*(k-L-N_m)$.

Применение линейного предсказания в FAP-алгоритмах основано на представлении матриц, формирующих матрицу $\mathbf{R}_L(k)$, в клеточной форме и использовании лемм об обращении таких матриц [1–4]. Результатом этих вычислений являются энергии минимального значения ошибок линейного предсказания $E^f(k)$, $E^b(k)$ и векторы весовых коэффициентов фильтров линейного предсказания $\mathbf{h}_{L-1}^f(k)$, $\mathbf{h}_{L-1}^b(k)$, используемые при вычислении вектора $\boldsymbol{\varepsilon}_L(k)$ без применения матричных операций. Как следствие, AP-алгоритм становится быстрым, т.е. вычислительно эффективным, т.к. не содержит матричных операций.

Некоторые процедуры линейного предсказания быстрых RLS-алгоритмов со скользящим окном могут быть найдены в [14, 15].

Ниже приводится пример такой процедуры, используемой в одноканальном стабилизированном FAEST-алгоритме со скользящим окном, адаптированный для рассматриваемого многоканального FAP-алгоритма.

For $m=1,2,\dots,M$

$$\alpha_m^{fU}(k) = x_m^*(k) - \mathbf{h}_{L-1}^{fDH}(k-1) \mathbf{x}_{L-1,m}^*(k-1)$$

$$e_m^{fU}(k) = \alpha_m^{fU}(k) \varphi_m^U(k-1)$$

$$\eta_m^U(k) = \frac{\alpha_m^{fU}(k)}{E^{fD}(k-1)}$$

$$\begin{bmatrix} \tilde{\mathbf{q}}_{L-1,m}^U(k) \\ \tilde{q}_m^U(k) \end{bmatrix} = \begin{bmatrix} 0 \\ \mathbf{t}_{L-1,m}^U(k-1) \end{bmatrix} + \begin{bmatrix} 1 \\ -\mathbf{h}_{L-1,m}^{fD}(k-1) \end{bmatrix} \eta_m^U(k)$$

$$\mathbf{h}_{L-1}^{fU}(k) = \mathbf{h}_{L-1}^{fD}(k-1) + \mathbf{t}_{L-1,m}^U(k-1) e_m^{fU*}(k)$$

$$E^{fU}(k) = E^{fD}(k-1) + e_m^{fU}(k) \alpha_m^{fU*}(k)$$

$$\frac{1}{\gamma_m^U(k)} = \frac{1}{\varphi_m^U(k-1)} + \alpha_m^{fU*}(k) \eta_m^U(k)$$

$$\alpha_m^{bU}(k) = x_m^*(k-L) - \mathbf{h}_{L-1}^{bDH}(k-1) \mathbf{x}_{L-1,m}^*(k)$$

$$q_m^U(k) = \frac{\alpha_m^{bU}(k)}{E^{bD}(k-1)}$$

$$\tilde{\alpha}_m^{bU}(k) = E^{bD}(k-1) \tilde{q}_m^U(k)$$

$$\alpha_m^{b(1)U}(k) = K_1 \alpha_m^{bU}(k) + (1-K_1) \tilde{\alpha}_m^{bU}(k)$$

$$\alpha_m^{b(2)U}(k) = K_2 \alpha_m^{bU}(k) + (1-K_2) \tilde{\alpha}_m^{bU}(k)$$

$$\alpha_m^{b(5)U}(k) = K_5 \alpha_m^{bU}(k) + (1-K_5) \tilde{\alpha}_m^{bU}(k)$$

$$t_m^U(k) = K_4 q_m^U(k) + (1-K_4) \tilde{q}_m^U(k)$$

$$\mathbf{t}_{L-1,m}^U(k) = \tilde{\mathbf{q}}_{L-1,m}^U(k) + \mathbf{h}_{L-1}^{bD}(k-1) t_m^U(k)$$

$$\frac{1}{\hat{\varphi}_m^U(k)} = 1 + \mathbf{t}_{L-1,m}^{UH}(k) \mathbf{x}_{L-1,m}^*(k)$$

$$\frac{1}{\tilde{\varphi}_m^U(k)} = \frac{1}{\gamma_m^U(k)} - \alpha_m^{b(5)U*}(k) \tilde{q}_m^U(k)$$

$$\varphi_m^U(k) = \left[K_3 \frac{1}{\hat{\varphi}_m^U(k)} + (1-K_3) \frac{1}{\tilde{\varphi}_m^U(k)} \right]^{-1}$$

$$e_m^{b(1)U}(k) = \alpha_m^{b(1)U}(k) \tilde{\varphi}_m^U(k)$$

$$e_m^{b(2)U}(k) = \alpha_m^{b(2)U}(k) \tilde{\varphi}_m^U(k)$$

$$E^{bU}(k) = E^{bD}(k-1) + e_m^{b(2)U}(k) \alpha_m^{b(2)U*}(k)$$

$$\mathbf{h}_{L-1}^{bU}(k) = \mathbf{h}_{L-1}^{bD}(k-1) + \mathbf{t}_{L-1,m}^U(k) e_m^{b(1)U*}(k)$$

$$\alpha_m^{fD}(k) = x_m^*(k-N_m) -$$

$$-\mathbf{h}_{L-1,m}^{fUH}(k) \mathbf{x}_{L-1,m}^*(k-1-N_m)$$

$$e_m^{fD}(k) = \alpha_m^{fD}(k) \phi_m^D(k-1)$$

$$\eta_m^D(k) = \frac{\alpha_m^{fD}(k)}{E^{fU}(k)}$$

$$\begin{bmatrix} \tilde{\mathbf{q}}_{L-1,m}^D(k) \\ \tilde{\mathbf{q}}_m^D(k) \end{bmatrix} = \begin{bmatrix} 0 \\ \mathbf{t}_{L-1,m}^D(k-1) \end{bmatrix} + \begin{bmatrix} 1 \\ -\mathbf{h}_{L-1,m}^{fU}(k) \end{bmatrix} \eta_m^D(k)$$

$$\mathbf{h}_{L-1}^{fD}(k) = \mathbf{h}_{L-1}^{fU}(k) - \mathbf{t}_{L-1,m}^D(k-1) e_m^{fD*}(k)$$

$$E^{fD}(k) = E^{fU}(k) - e_m^{fD}(k) \alpha_m^{fD*}(k)$$

$$\frac{1}{\gamma_m^D(k)} = \frac{1}{\phi_m^D(k-1)} - \alpha_m^{fD*}(k) \eta_m^D(k)$$

$$\phi_m^D(k) = \left[K_3 \frac{1}{\hat{\phi}_m^D(k)} + (1-K_3) \frac{1}{\tilde{\phi}_m^D(k)} \right]^{-1}$$

$$\alpha_m^{bD}(k) = x_m^*(k-L-N_m) - \mathbf{h}_{L-1,m}^{bUH}(k) \mathbf{x}_{L-1,m}^*(k-N_m)$$

$$q_m^D(k) = \frac{\alpha_m^{bD}(k)}{E^{bU}(k)}$$

$$\tilde{\alpha}_m^{bD}(k) = E^{bU}(k) \tilde{q}_m^D(k)$$

$$\alpha_m^{b(1)D}(k) = K_1 \alpha_m^{bD}(k) + (1-K_1) \tilde{\alpha}_m^{bD}(k)$$

$$\alpha_m^{b(2)D}(k) = K_2 \alpha_m^{bD}(k) + (1-K_2) \tilde{\alpha}_m^{bD}(k)$$

$$\alpha_m^{b(5)D}(k) = K_5 \alpha_m^{bD}(k) + (1-K_5) \tilde{\alpha}_m^{bD}(k)$$

$$t_m^U(k) = K_4 q_m^U(k) + (1-K_4) \tilde{q}_m^U(k)$$

$$\mathbf{t}_{L-1,m}^D(k) = \tilde{\mathbf{q}}_{L-1,m}^D(k) + \mathbf{h}_{L-1}^{bU}(k) t_m^D(k)$$

$$\frac{1}{\hat{\phi}_m^D(k)} = 1 - \mathbf{t}_{L-1,m}^{DH}(k) \mathbf{x}_{L-1,m}^*(k-N_m)$$

$$\frac{1}{\tilde{\phi}_m^D(k)} = \frac{1}{\gamma_m^D(k)} + \alpha_m^{b(5)D*}(k) \tilde{q}_m^D(k)$$

$$e_m^{b(1)D}(k) = \alpha_m^{b(1)D}(k) \tilde{\phi}_m^D(k)$$

$$e_m^{b(2)D}(k) = \alpha_m^{b(2)D}(k) \tilde{\phi}_m^D(k)$$

$$E^{bD}(k) = E^{bU}(k) - e_m^{b(2)D}(k) \alpha_m^{b(2)D*}(k)$$

$$\mathbf{h}_{L-1}^{bD}(k) = \mathbf{h}_{L-1}^{bU}(k) - \mathbf{t}_{L-1,m}^D(k) e_m^{b(1)D*}(k)$$

$$\phi_m^U(k-1) = \phi_m^U(k); \phi_m^D(k-1) = \phi_m^D(k);$$

$$E^{fD}(k-1) = E^{fD}(k); E^{bD}(k-1) = E^{bD}(k);$$

$$\mathbf{h}_{L-1}^{fD}(k-1) = \mathbf{h}_{L-1}^{fD}(k); \mathbf{h}_{L-1}^{bD}(k-1) = \mathbf{h}_{L-1}^{bD}(k)$$

End for m

$$\mathbf{h}_{L-1}^f(k) = \mathbf{h}_{L-1}^{fD}(k), \mathbf{h}_{L-1}^b(k) = \mathbf{h}_{L-1}^{bD}(k)$$

$$E^f(k) = E^{fD}(k), E^b(k) = E^{bD}(k),$$

Здесь верхние индексы f и b указывают на вычисления, относящиеся к предсказанию вперед (forward) и назад (backward) сигналов $x_m^*(k)$, $x_m^*(k-N_m)$, $x_m^*(k-L)$ и $x_m^*(k-L-N_m)$, а K_1, \dots, K_5 – коэффициенты стабилизации алгоритма линейного предсказания. Параметр экспоненциального забывания, обычно используемый в RLS-алгоритмах, здесь отсутствует, так как линейное предсказание осуществляется для M сигналов с помощью M фильтров с одинаковым числом весовых коэффициентов, равным L , на прямоугольных скользящих окнах данных в N_m отсчётов, что обусловлено структурой матрицы $\mathbf{X}_{NL}(k)$.

При использовании в FAP-алгоритме приведенной процедуры линейного предсказания требуется дополнительно инициализировать переменные

$$\mathbf{h}_{L-1}^{fD}(0) = \mathbf{0}_{L-1}; \mathbf{h}_{L-1}^{bD}(0) = \mathbf{0}_{L-1}; E^{fD}(0) = \delta^2;$$

$$E^{bD}(0) = \delta^2; \mathbf{t}_{L-1,m}^U(0) = \mathbf{0}_{L-1}; \mathbf{t}_{L-1,m}^D(0) = \mathbf{0}_{L-1};$$

$$\phi_m^U(k) = 1; \phi_m^D(0) = 1; m = 1, \dots, M.$$

V. ВЫЧИСЛЕНИЕ ПЕРЕМЕННОГО ШАГА СХОДИМОСТИ

Используя [16] в настоящей работе получено уравнение для переменного шага сходимости, изменяемого по градиентному закону (Gradient Adaptive Step-Size, GASS, со своим шагом сходимости ρ), вычисляемого на каждой итерации рассматриваемого многоканального FAP-алгоритма. Это уравнение имеет вид:

$$\begin{aligned} \mu(k) &= \mu(k-1) - \rho \frac{\partial \alpha^2(k)}{\partial \mu(k-1)} = \mu(k-1) + \\ &+ 2\rho \operatorname{Re} \left[\alpha^*(k) \boldsymbol{\varepsilon}_L^H(k-1) \mathbf{X}_{NL}^H(k-1) \mathbf{x}_N(k) \right] = \\ &= \mu(k-1) + 2\rho \operatorname{Re} \left[\alpha^*(k) \boldsymbol{\varepsilon}_L^H(k-1) \mathbf{r}_L(k) \right]. \end{aligned}$$

Допустимые границы значений параметра ρ определить аналитически не представляется возможным. Поэтому значение ρ обычно определяется экспериментально.

При вычислении $\mu(k)$ в FAP-алгоритме требуется инициализировать вектор $\boldsymbol{\varepsilon}_L(0) = \mathbf{0}_L$ и вычислять вектор $\mathbf{r}_L(k)$, содержащий L элементов. Если используется постоянный шаг сходимости $\mu(k) = \text{const}$, то указанная инициализация не требуется, а вектор $\mathbf{r}_{L-1}(k)$, может содержать лишь $L-1$ элемент. Так как $\mu(k)$ по определению может принимать только положительные действительные значения, то на каждой итерации его необходимо контролировать как $\mu_{\min} \leq \mu(k) \leq \mu_{\max}$, где μ_{\max} и μ_{\min}

– допустимые положительные значения шага сходимости, гарантирующие устойчивость FAP-алгоритма и малую дополнительную среднеквадратичную ошибку в установившемся состоянии.

VI. ВЫЧИСЛЕНИЕ ВЫХОДНОГО СИГНАЛА

FAP-алгоритм [6, 7] не может быть использован напрямую в задачах выравнивания амплитудно-частотной характеристики канала связи, предсказания сигналов в усилителях мощности, активного шумоподавления, в микрофонных решетках и ряде других задач, где требуется вычисление на каждой итерации не только сигнала ошибки $\alpha(k)$, но и выходного сигнала $y(k) = \mathbf{h}_N^H(k-1)\mathbf{x}'_N(k)$ адаптивного фильтра в явном виде. Для этого необходимо знать вектор весовых коэффициентов $\mathbf{h}_N(k)$, который в FAP-алгоритме не вычисляется. Эта проблема в FAP-алгоритме для многоканального адаптивного фильтра может быть решена аналогично [17]. Для этого требуется использовать уравнение, которое определяет неиспользуемый в явном виде в адаптивном фильтре на базе FAP-алгоритма вектор весовых коэффициентов

$$\mathbf{h}_N(k) = \hat{\mathbf{h}}_N(k) + \bar{\mathbf{X}}_{N(L-1)}^H(k)\bar{\mathbf{p}}_{L-1}(k),$$

где матрица $\bar{\mathbf{X}}_{N(L-1)}^H(k)$ состоит из первых L столбцов матрицы $\mathbf{X}_{NL}(k)$ [6]. С помощью этого уравнения выходной сигнал адаптивного фильтра вычисляется с линейной сложностью как

$$\begin{aligned} y(k) &= \mathbf{h}_N^H(k-1)\mathbf{x}'_N(k) = \hat{\mathbf{h}}_N^H(k-1)\mathbf{x}'_N(k) + \\ &+ \bar{\mathbf{p}}_{L-1}^H(k-1)\bar{\mathbf{X}}_{N(L-1)}^H(k-1)\mathbf{x}'_N(k) = \hat{\mathbf{h}}_N^H(k-1)\mathbf{x}'_N(k) + \\ &+ \bar{\mathbf{p}}_{L-1}^H(k-1)\mathbf{r}'_{L-1}(k), \end{aligned}$$

где

$$\begin{aligned} \mathbf{x}'_N(k) &= [\mathbf{x}'_{N_1}(k), \mathbf{x}'_{N_2}(k), \dots, \mathbf{x}'_{N_M}(k)]^T, \\ \mathbf{x}'_{N_m}(k) &= [x'_m(k), x'_m(k-1), \dots, x'_m(k-N_m+1)]^T, \\ \mathbf{r}'_{L-1}(k) &= \bar{\mathbf{X}}_{N(L-1)}^H(k-1)\mathbf{x}'_N(k) = \\ &= \mathbf{r}'_{L-1}(k-1) + \sum_{m=1}^M [x'_m(k)\mathbf{x}_{L-1,m}^*(k-1) - \\ &- x'_m(k-N_m)\mathbf{x}_{L-1,m}^*(k-N_m-1)]. \end{aligned}$$

Эти уравнения приведены для более общего случая, возникающего в задачах активного шумоподавления и предсказания сигналов в усилителях мощности, где весовые коэффициенты адаптивного фильтра рассчитываются, используя сигналы $x_m(k)$, а выходной сигнал рассчитывается, используя сигналы $x'_m(k) \neq x_m(k)$.

В задачах выравнивания амплитудно-частотной характеристики канала связи и адаптации микрофонных решеток указанные сигналы равны, т.е.

$x'_m(k) = x_m(k)$, а также равны векторы $\mathbf{r}'_{L-1}(k)$ и $\mathbf{r}_{L-1}(k)$. В этом случае

$$\begin{aligned} y(k) &= \mathbf{h}_N^H(k-1)\mathbf{x}_N(k) = \hat{\mathbf{h}}_N^H(k-1)\mathbf{x}_N(k) + \\ &+ \bar{\mathbf{p}}_{L-1}^H(k-1)\bar{\mathbf{X}}_{N(L-1)}^H(k-1)\mathbf{x}_N(k) = \hat{\mathbf{h}}_N^H(k-1)\mathbf{x}_N(k) + \\ &+ \bar{\mathbf{p}}_{L-1}^H(k-1)\mathbf{r}_{L-1}(k). \end{aligned}$$

VII. ВЫЧИСЛИТЕЛЬНАЯ СЛОЖНОСТЬ БЫСТРОГО АЛГОРИТМА АФФИННЫХ ПРОЕКЦИЙ

Без учета процедуры линейного предсказания и вычисления переменных $\mu(k)$ и $y(k)$ оценка вычислительной сложности основной процедуры FAP-алгоритма равна $2N + (2M + 8)L$ операциям умножения, $2N + (2M + 6)L$ операциям сложения и двум операциям деления, т.е. является линейной функцией параметров N , M и L .

Сложность вычисления переменных $E^f(k)$, $E^b(k)$, $\mathbf{h}_{L-1}^f(k)$ и $\mathbf{h}_{L-1}^b(k)$ зависит от используемой процедуры линейного предсказания и может составлять $(10 \dots 20)ML$ операций умножения и $(10 \dots 20)ML$ операций сложения. Для приведенной в настоящей работе процедуры эта оценка составляет $14ML$ операций умножения, $14ML$ операций сложения и $6M$ операций деления.

Вычисление переменного шага сходимости $\mu(k)$ увеличивает сложность многоканального FAP-алгоритма незначительно. Требуется лишь дополнительно L умножений и L сложений при вычислении скалярного произведения $\mathbf{e}_L^H(k-1)\mathbf{r}_L(k)$, так как первые $L-1$ элементов используемого здесь вектора $\mathbf{r}_L(k)$ вычисляются в основной процедуре FAP-алгоритма, а вычисление L -го элемента требует лишь $2M$ умножений и $2M$ сложений.

Сложность вычисления выходного сигнала адаптивного фильтра $y(k)$ равна $N + (2M + 1)L$ операциям умножения и $N + (2M + 1)L$ операциям сложения. Если $x'_m(k) = x_m(k)$, то дополнительно вычислять $\mathbf{r}'_{L-1}(k) = \mathbf{r}_{L-1}(k)$ не требуется, что уменьшает сложность определения $y(k)$.

Таким образом, если $M \ll N$, $L \ll N$ и $M \ll L$, то вычислительная сложность многоканальных FAP- и NLMS-алгоритмов сравнима и является линейной функцией полного числа весовых коэффициентов N адаптивного фильтра. Она оценивается как примерно $2N$ операций умножения и $2N$ операций сложения.

Рассмотренные оценки показывают число комплексных или действительных операций в зависимости от используемой арифметики.

Версия FAP-алгоритма для адаптивного фильтра с действительными весовыми коэффициентами

образуется путем исключения операций комплексного сопряжения переменных и применения арифметики действительных чисел. Кроме того, в уравнениях

$$\mathbf{a}_L(k) = \begin{bmatrix} \alpha(k) \\ [1 - 2\mu(k-1)]\bar{\mathbf{a}}_{L-1}(k-1) \end{bmatrix},$$

$$\mathbf{s}_{L-1}(k) = [1 - 2\mu(k)]\mathbf{z}_{L-1}(k),$$

$$\mathbf{p}_L(k) = 2\mu(k)\mathbf{e}_L(k) + \begin{bmatrix} 0 \\ \bar{\mathbf{p}}_{L-1}(k-1) \end{bmatrix}$$

рядом с переменными $\mu(k-1)$ и $\mu(k)$ появляется коэффициент, равный двум.

VIII. МОДЕЛИРОВАНИЕ

Работоспособность и эффективность рассмотренного многоканального FAP-алгоритма подтверждена путем компьютерного моделирования задачи активного шумоподавления [11, 18]. Параметры адаптивного фильтра были выбраны как $N_1 = N_2 = 350$, $L = 4$, $\mu(0) = 10^{-3}$, $\mu_{\max} = 2\mu(0)$, $\mu_{\min} = 10^{-7}$, $\rho = 10^{-5}$, $\delta^2 = 10^{-3}$. Результаты моделирования показаны на рис. 2.

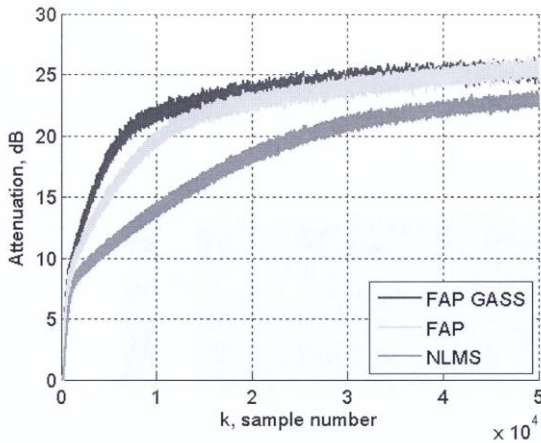


Рис. 2. Результаты моделирования: ослабление шума

Здесь представлены графики достигаемого подавления (Attenuation) акустического шума. Сравнивались три адаптивных алгоритма: NLMS-алгоритм (тоже самое что AP алгоритм при $L=1$), FAP-алгоритм и GASS FAP-алгоритм. Из рис. 2 следует, что в данной задаче NLMS-алгоритм сходится значительно медленнее, чем FAP-алгоритм. Использование переменного шага сходимости также ускоряет сходимость FAP-алгоритма.

IX. ЗАКЛЮЧЕНИЕ

Таким образом, рассмотренный многоканальный FAP-алгоритм может быть использован в приложениях, где требуются быстрая сходимость адаптивного фильтра, формирование физического выходного сигнала и имеются ограничения на ресурсы для реализации.

ЛИТЕРАТУРА

- [1] Haykin S. Adaptive filter theory. 5-th edition. Pearson Education, 2014. 889 p.
- [2] Diniz P.S.R. Adaptive filtering. Algorithms and practical implementation. 4-th edition. Springer, 2013. 652 p.
- [3] Farhang-Boroujeny B. Adaptive filters. Theory and applications. 2-nd edition. John Wiley and Sons, 2013. 778 p.
- [4] Джиган В.И. Адаптивная фильтрация сигналов: теория и алгоритмы. М.: Техносфера, 2013. 528 с.
- [5] Ozeki K., Umeda K. An adaptive filtering algorithm using orthogonal projection to an affine subspace and its properties // Trans. IECE Japan, 1984. Vol. J67-A. № 2. P. 126–132.
- [6] Gay S.L. A fast converging, low complexity adaptive filtering algorithm // Proc. of the 3-rd International Workshop on Acoustic Echo Control. 1993. P. 223–226.
- [7] Tanaka M., Kaneda Y., Makino S., Kojima J. A fast projection algorithm for adaptive filtering // IEICE Transactions. Fundamentals. 1995. Vol. E78A. № 10. P. 1355–1361.
- [8] Messerschmitt D. Echo cancellation in speech and data transmission // IEEE Journal on Selected Areas in Communications. 1984. Vol. 2. № 2. P. 283–297.
- [9] Makino S. Acoustic echo cancellation // IEEE Signal Processing Magazine. 1997. Vol. 14. № 5. P. 39–41.
- [10] Qureshi S. Adaptive equalization // Proceedings of the IEEE. 1985. Vol. 73. № 9. P. 1349–1387.
- [11] Kuo S.M., Morgan D.S. Active noise control // Proceedings of the IEEE. 1999. Vol. 87. № 6. P. 943–973.
- [12] Saleh A.A.M., Salz J. Adaptive linearization of power amplifiers in digital radio systems // The Bell System Technical Journal. 1983. Vol. 62. № 4. P. 1019–1033.
- [13] Benesty J., Huang Y. Adaptive signal processing. Application to real-world problems. Springer, 2003. 356 p.
- [14] Cioffi J.M., Kailath T. Windowed fast transversal filters adaptive algorithms with normalization // IEEE Trans. Acoustics, Speech, and Signal Processing. 1985. Vol. 33. № 3. P. 607–625.
- [15] Джиган В.И. Многоканальные RLS- и быстрые RLS-алгоритмы адаптивной фильтрации // Успехи современной радиоэлектроники. 2004. № 11. С. 48–77.
- [16] Djigan V.I. Improved fast affine projection algorithm with gradient adaptive step-size // Proceedings of the 3-rd International Conference on Antennas, Radio-communication Systems & Mean. 1997. Vol. 3. P. 23–32.
- [17] Douglas S.C. The fast affine projection algorithm for active noise control. Conference Record of the Twenty-Ninth Asilomar Conference. 1995. Vol. 2. P. 1245–1249.
- [18] Active Noise Control, Mathworks® Web Resource, <http://www.mathworks.com/help/dsp/examples/active-noise-control-using-a-filtered-x-lms-fir-adaptive-filter.html#zmw57dd0e3970> (accessed 27.03.2016).

Fast Affine Projection Algorithm: Full Version

V.I. Djigan

National Research University of Electronic Technology, djigan@yandex.ru

Keywords — multichannel adaptive filter, fast affine projection algorithm, variable step-size, output signal calculation.

ABSTRACT

The fast affine projection algorithm for multichannel adaptive filters with complex-valued weights is described. The architecture of such filter is presented. The filter may have different number of weights in its channels.

The general form of the affine projection algorithm is shown. The normalization matrix of the algorithm consists of the sum of such matrices over all channels. The matrices have the same dimension that is a square of the projection size. Each of the matrices is evaluated over an own sliding window of the samples which corresponds the number of the weights in the proper channels of the adaptive filter.

The full computational procedure of the fast version of the algorithm is presented. The procedure takes into consideration the above peculiarity of the multichannel adaptive filter.

The procedure also includes the linear prediction of input signals used for the fast computation of the adaptive filter normalized errors vector. The prediction is adopted from the sliding window stabilized fast recursive least squares algorithm, called «a fast posteriori error sequential technique».

Original version of the known single-channel fast affine projection algorithm does not compute the adaptive filter output signal explicitly. At the same time the signal is required in a number of applications of adaptive filters. It is shown how to compute the signal in the multichannel fast affine projection algorithm without the increasing of the algorithm complexity.

Besides a procedure of a variable step-size computation based on the gradient search strategy is also presented. It also does not increase the adaptive filtering algorithm complexity.

The computation complexity of the different parts of the fast affine projection algorithm is evaluated. According the evaluation the complexity is a linear function of the total number of weights in the adaptive filter channels, number of the channels and projection value.

The simulation of the two-channel adaptive filter in the active noise control task has demonstrated the developed algorithm efficiency. The simulation test-bench was adopted from the standard MATLAB example of an active control task.

REFERENCES

- [1] Haykin S. Adaptive filter theory, 5-th edition. Pearson Education, 2014, 889 p.
- [2] Diniz P.S.R. Adaptive filtering. Algorithms and practical implementation, 4-th edition. Springer, 2013, 652 p.
- [3] Farhang-Boroujeny B. Adaptive filters. Theory and applications, 2-nd edition, John Wiley and Sons, 2013, 778 p.
- [4] Djigan V.I. Adaptivnaja fil'tracija signalov: teorija i algoritmy – Adaptive signal filtering: theory and algorithms, Moscow, Technosfera, 2013, 528 p. (in Russian).
- [5] Ozeki K., Umeda K. An adaptive filtering algorithm using orthogonal projection to an affine subspace and its properties. Trans. IECE Japan, 1984, vol. J67-A, no. 2, pp. 126-132.
- [6] Gay S.L. A fast converging, low complexity adaptive filtering algorithm. Proc. of the 3-rd International Workshop on Acoustic Echo Control, 1993, pp. 223-226.
- [7] Tanaka M., Kaneda Y., Makino S., Kojima J. A fast projection algorithm for adaptive filtering. IEICE Transactions. Fundamentals, 1995, vol. E78A, no. 10, pp. 1355-1361.
- [8] Messerschmitt D. Echo cancellation in speech and data transmission. IEEE Journal on Selected Areas in Communications, 1984, vol. 2, no.2, pp. 283-297.
- [9] Makino S. Acoustic echo cancellation. IEEE Signal Processing Magazine, 1997, vol. 14, no. 5, pp. 39-41.
- [10] Qureshi S. Adaptive equalization. Proceedings of the IEEE, 1985, vol. 73, no. 9, pp. 1349-1387.
- [11] Kuo S.M., Morgan D.S. Active noise control. Proceedings of the IEEE, 1999, vol. 87, no. 6, pp. 943-973.
- [12] Saleh A.A.M., Salz J. Adaptive linearization of power amplifiers in digital radio systems. The Bell System Technical Journal, 1983, vol. 62, no. 4, pp. 1019-1033.
- [13] Benesty J., Huang Y. Adaptive signal processing. Application to real-world problems, Springer, 2003, 356 p.
- [14] Cioffi J.M., Kailath T. Windowed fast transversal filters adaptive algorithms with normalization. IEEE Trans. Acoustics, Speech, and Signal Processing, 1985, vol. 33, no. 3, pp. 607-625.
- [15] Djigan V.I. Multichannel RLS and fast RLS adaptive filtering algorithms. Uspehi Sovremennoj Radioelektroniki, Moscow, Russia, 2004, No. 11, pp. 48-77. (in Russian).
- [16] Djigan V.I. Improved fast affine projection algorithm with gradient adaptive step-size. Proceedings of the 3-rd International Conference on Antennas, Radio-communication Systems & Mean, 1997, vol. 3, pp. 23-32.
- [17] Douglas S.C. The fast affine projection algorithm for active noise control. Conference Record of the Twenty-Ninth Asilomar Conference, 1995, vol. 2, pp. 1245-1249.
- [18] Active Noise Control, Mathworks® Web Resource, <http://www.mathworks.com/help/dsp/examples/active-noise-control-using-a-filtered-x-lms-fir-adaptive-filter.html#zmw57dd0e3970> (accessed 27.03.2016).