

# LMS-алгоритм адаптивной фильтрации сигналов: первый или единственный для использования на практике?

В.И. Джиган

ООО «Техкомпания Хуавэй», victor.dzhigan@huawei.com

**Аннотация** — В статье рассматривается LMS-алгоритм адаптивной фильтрации, широко используемый в различных устройствах с момента его изобретения до настоящего времени. Показаны основные шаги получения этого алгоритма и его свойства. Приводятся нормализованная, с переменным шагом сходимости и линейно-ограниченная версии LMS-алгоритма. Обсуждаются программные средства моделирования и реализация этого алгоритма на современных FPGA и цифровых сигнальных процессорах.

**Ключевые слова** — Least Mean Square (LMS), Normalized Least Mean Square (NLMS), метод наименьшего квадрата (МНК), переменный шаг сходимости, линейные ограничения.

## I. ВВЕДЕНИЕ

Адаптивная фильтрация является одним из широко востребованных научно-технических направлений в современной цифровой обработке сигналов (ЦОС). Зарождению и становлению этого направления предшествовали достижения в математике, радиотехнике и связи [1]. Однако на практике адаптивная фильтрация сигналов начала применяться с момента появления самого простого адаптивного фильтра по критерию наименьшего квадрата (Least Mean Square, LMS). Это гениальное изобретение, сделанное в конце 1950-х годов, принадлежит профессору Стэнфордского университета (США) Бернарду Уидроу [2 – 4]. В российской литературе для обозначения LMS-алгоритма часто используется аббревиатура МНК, что означает метод наименьшего квадрата.

Несмотря на не самую высокую эффективность по сравнению с более сложными рекурсивными алгоритмами по критерию наименьших квадратов (Recursive Least Squares, RLS), LMS-алгоритм вычисления весовых коэффициентов (ВК) адаптивных фильтров и сегодня является самым широко используемым на практике. Причинами успеха LMS-алгоритма являются его низкая вычислительная сложность (наименьшая среди всех алгоритмов адаптивной фильтрации), простота в понимании и реализации, а также устойчивость при правильном выборе единственного параметра, именуемого шагом сходимости. Вычислительная сложность LMS-алгоритма равна  $2N$  арифметическим операциям (сложений с умножениями) на одну итерацию,

совпадающую по длительности с периодом дискретизации обрабатываемых сигналов, где  $N$  – число ВК адаптивного фильтра.

LMS-алгоритм относится к классу алгоритмов стохастического градиентного поиска координат минимума целевой функции. Термин «стохастический» употребляется для того, чтобы отличать LMS-алгоритм, использующий упрощенный градиент, от алгоритма наискорейшего спуска, использующего точное значение градиента целевой функции (среднеквадратической ошибки, Mean Square Error, MSE), минимизируемой в процессе работы адаптивного фильтра [4 – 6]. Упрощенный градиент является неточным, в результате чего движение вектора ВК к минимуму этой функции носит в некотором смысле случайный характер, отчего и используется термин «стохастический».

Несмотря на солидный возраст, LMS-алгоритм до сих пор является предметом пристального внимания тысяч исследователей и инженеров во всем мире. По разным оценкам, не менее 50% всех публикаций в области адаптивной фильтрации рассматривают различные аспекты этого алгоритма и его разновидностей, а на практике он используется не менее, чем в 80% случаев. Однако опыт общения со многими инженерами, использующими LMS-алгоритм в своих разработках, показывает, что эта простота часто приводит к неправильному пониманию его свойств, а, значит, к неэффективному функционированию, а иногда и к неработоспособности.

Настоящая статья освещает основные аспекты LMS-алгоритма, которые, по мнению автора, могут способствовать более правильному его пониманию и использованию. Рассматриваются основные разновидности, свойства и вычислительные процедуры данного алгоритма, а также современные программные средства моделирования, которые позволяют изучать его особенности и работу в приложениях. Кроме того, приводятся примеры реализации LMS-алгоритма в современных программируемых вентильных матрицах (Field-Programmable Gate Array, FPGA) и цифровых сигнальных процессорах (ЦСП) отечественного и зарубежного производства.

## II. ВЫЧИСЛИТЕЛЬНАЯ ПРОЦЕДУРА LMS-АЛГОРИТМА

Один из способов получения вычислительной процедуры LMS-алгоритма адаптивной фильтрации заключается в следующем [4–6]. Вектор ВК адаптивного фильтра (винеровское решение задачи оптимальной линейной фильтрации сигналов), определяемый как  $\mathbf{h}_{N,o} = \mathbf{R}_N^{-1} \mathbf{r}_N$ , соответствует минимальной среднеквадратической ошибке оценивания требуемого сигнала  $d(k)$  этого фильтра в предположении, что его входной сигнал  $x(k)$  и сигнал  $d(k)$  стационарны. Здесь  $\mathbf{R}_N = E\{\mathbf{x}_N(k)\mathbf{x}_N^H(k)\}$  – корреляционная матрица входного сигнала адаптивного фильтра, а  $\mathbf{r}_N = E\{\mathbf{x}_N(k)d^*(k)\}$  – вектор взаимной корреляции этого сигнала и сигнала  $d(k)$ .

Если на  $k$ -м временном отсчете имеются оценки  $\hat{\mathbf{R}}_N(k)$  и  $\hat{\mathbf{r}}_N(k)$  этой матрицы и этого вектора, то для поиска винеровского решения можно использовать алгоритм наискорейшего спуска

$$\begin{aligned} \mathbf{h}_N(k) &= \mathbf{h}_N(k-1) - \mu \hat{\nabla}_{\mathbf{h}_N} \xi(k-1) = \\ &= \mathbf{h}_N(k-1) + \mu \hat{\mathbf{r}}_N(k) - \mu \hat{\mathbf{R}}_N(k) \mathbf{h}_N(k-1), \end{aligned}$$

где  $\hat{\nabla}_{\mathbf{h}_N} \xi(k-1) = -\hat{\mathbf{r}}_N(k) + \hat{\mathbf{R}}_N(k) \mathbf{h}_N(k-1)$  – оценка градиента целевой функции, а  $\mu$  – шаг сходимости.

Если оценивать  $\mathbf{R}_N$  и  $\mathbf{r}_N$  на окне всего в один отсчет обрабатываемых сигналов, т. е. как  $\hat{\mathbf{R}}_N(k) = \mathbf{x}_N(k)\mathbf{x}_N^H(k)$  и  $\hat{\mathbf{r}}_N(k) = \mathbf{x}_N(k)d^*(k)$ , то оценка градиента определяется как  $\hat{\nabla}_{\mathbf{h}_N} \xi(k-1) = -\mathbf{x}_N(k)\alpha^*(k)$ , где  $\alpha(k) = d(k) - \mathbf{x}_N(k)\mathbf{h}_N^H(k-1)$  – априорная ошибка моделирования сигнала  $d(k)$ . С учетом выражения  $\hat{\nabla}_{\mathbf{h}_N} \xi(k-1)$  уравнение вычисления вектора ВК адаптивного фильтра принимает вид

$$\mathbf{h}_N(k) = \mathbf{h}_N(k-1) + \mu \mathbf{x}_N(k)\alpha^*(k).$$

Это уравнение и уравнение вычисления ошибки  $\alpha(k)$  являются LMS-алгоритмом. Название Least Mean Square отражает свойство алгоритма минимизировать средний квадрат мгновенной ошибки, т. е. ошибки, определяемой по одному отсчету обрабатываемых сигналов.

Поскольку в этом уравнении используются неточные оценки градиента, то в адаптивном процессе возникает шум, в результате которого траектория движения вектора  $\mathbf{h}_N(k)$  к вектору  $\mathbf{h}_{N,o}$  не совпадает с аналогичной траекторией движения в алгоритме наискорейшего спуска.

Этот шум уменьшается по мере движения к вектору  $\mathbf{h}_{N,o}$ , так как приращения вектора ВК, равные  $-\mu \hat{\nabla}_{\mathbf{h}_N} \xi(k-1)$ , вблизи точки  $\mathbf{h}_{N,o}$  многомерного пространства уменьшаются по сравнению с такими приращениями в начальной точке  $\mathbf{h}_N(0) \neq \mathbf{h}_{N,o}$ . Это происходит из-за того, что компоненты градиента вблизи стационарной точки  $\mathbf{h}_{N,o}$  стремятся к нулю. Следовательно, и оценка этого градиента также стремится к нуль-вектору, т. е.  $\hat{\nabla}_{\mathbf{h}_N} \xi(k-1) \rightarrow \mathbf{0}_N$ .

При значении  $\mu$ , гарантирующем устойчивость, LMS-алгоритм обеспечивает сходимость к винеровскому решению (окрестности минимума MSE-функции). На рис. 1 показан пример поведения этого алгоритма в адаптивном фильтре с двумя ВК при разных начальных значениях ВК. Минимум MSE-функции показан крестиком в центре рисунка.

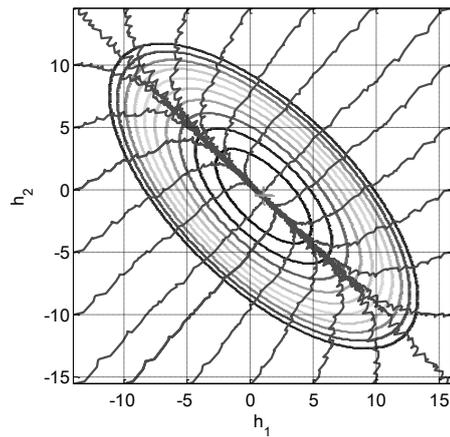


Рис. 1. Поведение LMS-алгоритма при различных начальных значениях ВК

## III. ВЫЧИСЛИТЕЛЬНАЯ СЛОЖНОСТЬ LMS-АЛГОРИТМА

Сложность LMS-алгоритма определяется следующим образом. Операция скалярного произведения векторов  $\mathbf{h}_N^H(k-1)\mathbf{x}_N(k)$  при вычислении ошибки  $\alpha(k)$  требует  $N$  умножений и  $N$  сложений на одну итерацию алгоритма. Если быть более точным, то число сложений равно  $N-1$ .

При программной реализации операция скалярного произведения векторов обычно выполняется как

$$\mathbf{h}_N^H(k-1)\mathbf{x}_N(k) = \sum_{n=1}^N h_n^*(k-1)x(k-n+1).$$

Начальное значение этой суммы задается равным нулю. Поэтому выражение

$$\begin{aligned} \sum_{n=1}^N h_n^*(k-1)x(k-n+1) &= \\ &= 0 + \underbrace{h_1^*(k-1)x(k) + \dots + h_N^*(k-1)x(k-N+1)}_{N \text{ умножений, } N \text{ сложений}} \end{aligned}$$

содержит  $N$  умножений и  $N$  сложений.

При аппаратной реализации, «лишняя» операция сложения с нулем обычно не выполняется, так как для этого требуется дополнительный сумматор. Поскольку адаптивные фильтры используются в приложениях, где, как правило,  $N \gg 1$ , то оценка числа сложений в операции  $\mathbf{h}_N^H(k-1)\mathbf{x}_N(k)$ , примерно равная  $N$ , является корректной и в случае аппаратной реализации этой операции.

Вычисление ВК представляет собой операцию умножения вектора на скаляр с векторным накоплением

$$\begin{aligned} \mathbf{h}_N(k) &= \mathbf{h}_N(k-1) + \gamma(k) \mathbf{x}_N(k) = \\ &= \begin{bmatrix} h_1(k-1) \\ h_2(k-1) \\ \vdots \\ h_N(k-1) \end{bmatrix} + \gamma(k) \begin{bmatrix} x_1(k) \\ x_2(k) \\ \vdots \\ x_N(k) \end{bmatrix} = \begin{bmatrix} h_1(k-1) + \gamma(k)x_1(k) \\ h_2(k-1) + \gamma(k)x_2(k) \\ \vdots \\ h_N(k-1) + \gamma(k)x_N(k) \end{bmatrix}, \end{aligned}$$

где  $\gamma(k) = \mu \alpha^*(k)$ . Это уравнение также требует  $N$  умножений и  $N$  сложений на одну итерацию, т. е. по одному умножению и одному сложению при вычислении нового значения каждого из  $N$  ВК, образующих вектор  $\mathbf{h}_N(k)$ .

Следовательно, оценка полной вычислительной сложности LMS-алгоритма равна  $2N$  арифметическим операциям.

#### IV. СВОЙСТВА LMS-АЛГОРИТМА

Так как в LMS-алгоритме значение градиента MSE-функции заменяется его оценкой, то траектория движения ВК (см. серые кривые переходных процессов примера на рис. 2) лишь в среднем совпадает с такой траекторией (см. темные кривые на рис. 2) в алгоритме наискорейшего спуска [1]. LMS-алгоритм также демонстрирует несколько большую среднеквадратическую ошибку  $\xi$  в установившемся состоянии (см. серую кривую примера переходного процесса на рис. 3), по сравнению с  $\xi_{\min}$  в алгоритме наискорейшего спуска (см. темную кривую на рис. 3). Величина дополнительной ошибки  $\xi_{\text{exc}}$  зависит от значений собственных чисел  $\lambda_n$  корреляционной матрицы входных сигналов адаптивного фильтра, шага сходимости LMS-алгоритма  $\mu$  и дисперсии аддитивного шума  $\sigma_0^2$ , обычно присутствующего на входе требуемого сигнала  $d(k)$ .

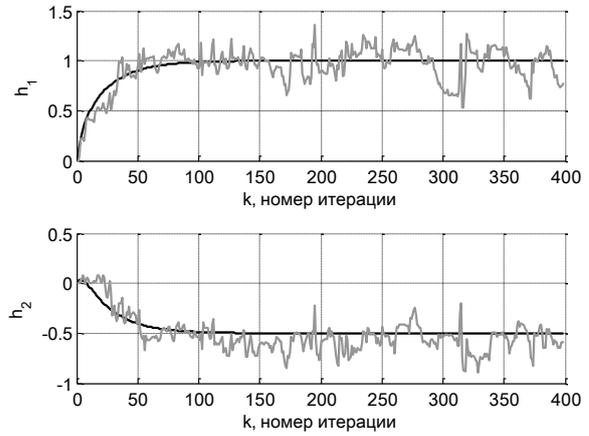


Рис. 2. Переходные процессы в терминах ВК

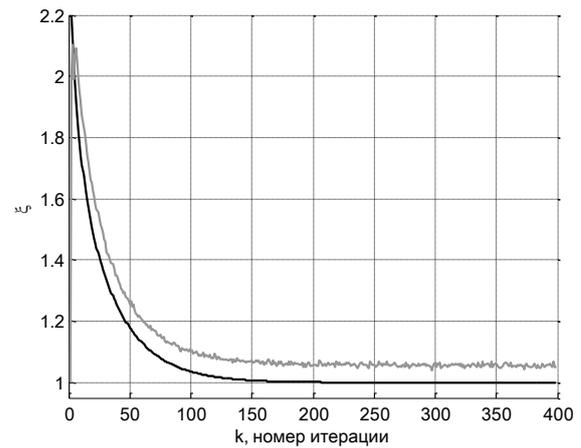


Рис. 3. Переходные процессы в терминах среднеквадратической ошибки

Качество адаптивной фильтрации с помощью LMS-алгоритма определяется относительным значением дополнительной среднеквадратической ошибки

$$M = \frac{\xi_{\text{exc}}}{\xi_{\min}} = \frac{\xi - \xi_{\min}}{\xi_{\min}}.$$

На практике шаг сходимости LMS-алгоритма обычно выбирается так, чтобы значение  $M$  не превышало 10%, что достигается, если  $\mu \ll 1$ .

В этом случае для адаптивного фильтра с комплексными ВК значение  $M$  определяется как

$$M \approx \frac{\mu \sum_{n=1}^N \lambda_n}{2 - \mu \sum_{n=1}^N \lambda_n} = \frac{\mu \text{tr}\{\mathbf{R}_N\}}{2 - \mu \text{tr}\{\mathbf{R}_N\}},$$

а для фильтра с действительными ВК – как

$$M \approx \frac{\mu \sum_{n=1}^N \lambda_n}{1 - \mu \sum_{n=1}^N \lambda_n} = \frac{\mu \operatorname{tr}\{\mathbf{R}_N\}}{1 - \mu \operatorname{tr}\{\mathbf{R}_N\}}.$$

Уравнение LMS-алгоритма для адаптивного фильтра с действительными ВК имеет вид

$$\mathbf{h}_N(k) = \mathbf{h}_N(k-1) + 2\mu \mathbf{x}_N(k)\alpha(k).$$

Подобно алгоритму наискорейшего спуска, устойчивость работы LMS-алгоритма определяется значением его шага сходимости и значениями собственных чисел корреляционной матрицы входного сигнала адаптивного фильтра. Так как поведение кривых изменения ВК LMS-алгоритма в среднем совпадает с поведением таких кривых алгоритма наискорейшего спуска (см. рис. 2), то часто в качестве шага сходимости, гарантирующего устойчивую работу LMS-алгоритма адаптивного фильтра с комплексными ВК, используют значение  $0 < \mu < \frac{2}{N\sigma_x^2}$ , определенное для алгоритма наискорейшего спуска. Для адаптивного фильтра с действительными ВК  $0 < \mu < \frac{1}{N\sigma_x^2}$ , где  $\sigma_x^2$  – дисперсия входного сигнала адаптивного фильтра.

На практике же, если значение шага сходимости приближается к максимальной границе, определенной указанным способом, адаптивный фильтр может становиться неустойчивым. Это обусловлено тем, что допустимое значение шага сходимости в LMS-алгоритме на самом деле несколько меньше, чем в алгоритме наискорейшего спуска. Согласно более строгому анализу [5, 6], адаптивный фильтр на базе LMS-алгоритма с комплексными ВК является устойчивым, если  $0 < \mu < \frac{1}{N\sigma_x^2}$ . Для фильтра с действительными ВК шаг сходимости должен быть ограничен как  $0 < \mu < \frac{1}{3N\sigma_x^2}$ .

Таким образом, предельно допустимое значение шага сходимости LMS-алгоритма меньше, чем алгоритма наискорейшего спуска. Это обусловлено тем, что в LMS-алгоритме используется не точное значение градиента целевой функции, а лишь его оценка.

Выражение для среднев квадратической ошибки  $\xi(k)$  LMS-алгоритма в среднем совпадает с аналогичным выражением для алгоритма наискорейшего спуска. Поэтому постоянные времени  $k_{\tau_{n,\xi}}$  переходного процесса кривой  $\xi(k)$  в обоих алгоритмах выражаются одинаковыми зависимостями, а именно, в случае комплексных ВК – как

$$\frac{1}{2\mu\lambda_{\max}} \leq k_{\tau_{n,\xi}} \leq \frac{1}{2\mu\lambda_{\min}},$$

а в случае действительных ВК – как

$$\frac{1}{4\mu\lambda_{\max}} \leq k_{\tau_{n,\xi}} \leq \frac{1}{4\mu\lambda_{\min}}.$$

Здесь  $\lambda_{\min}$  и  $\lambda_{\max}$  – минимальное и максимальное собственные числа корреляционной матрицы входного сигнала адаптивного фильтра. Данные соотношения, а также выражения для относительной дополнительной среднев квадратической ошибки  $M$ , объясняют поведение кривых  $\xi(k)$  (рис. 4).

В LMS-алгоритме при увеличении шага сходимости уменьшается длительность переходного процесса и увеличивается дополнительная среднев квадратическая ошибка. На рис. 4 приведены примеры двух переходных процессов при двух значениях шага сходимости и одном и том же входном сигнале, т. е. одинаковых корреляционных матрицах и их собственных числах. Шаг сходимости при получении кривой 1 меньше, чем шаг сходимости при получении кривой 2. Минимальное значение среднев квадратической ошибки на рис. 4 равно единице, что позволяет легко определить значение дополнительной среднев квадратической ошибки в каждом случае.

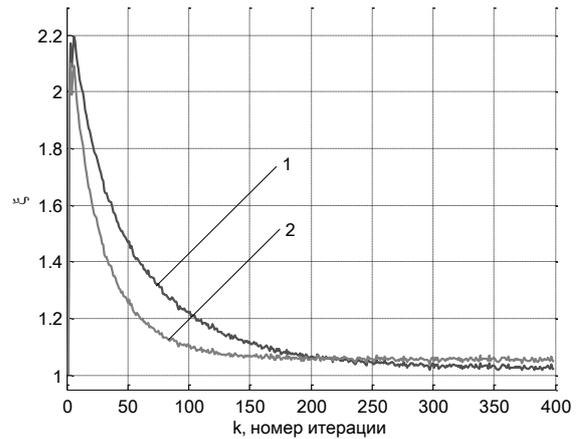


Рис. 4. Поведение LMS-алгоритма

Постоянная времени  $k_{\tau_{i,h}}$  переходного процесса среднего значения комплексных ВК LMS-алгоритма находится в пределах

$$\frac{1}{\mu\lambda_{\max}} \leq k_{\tau_{i,h}} \leq \frac{1}{\mu\lambda_{\min}},$$

а постоянная времени переходного процесса среднего значения действительных ВК – в пределах

$$\frac{1}{2\mu\lambda_{\max}} \leq k_{\tau_{n,h}} \leq \frac{1}{2\mu\lambda_{\min}},$$

т.е. эти пределы совпадают с аналогичными пределами для алгоритма наискорейшего спуска.

#### V. LMS-АЛГОРИТМ С ПЕРЕМЕННЫМ ШАГОМ СХОДИМОСТИ

LMS-алгоритм используется в приложениях, где можно априори оценить предельно допустимое значение шага сходимости  $\mu$ , гарантирующее устойчивость алгоритма. Такая оценка возможна, если известны статистические характеристики входного сигнала адаптивного фильтра. В приложениях, где эти характеристики неизвестны или меняются во времени (например, при обработке нестационарных сигналов), часто используется нормализованный LMS-алгоритм (Normalized LMS, NLMS).

Существует несколько способов получения вычислительной процедуры NLMS-алгоритма. Один из них заключается в том, что NLMS-алгоритм можно рассматривать как результат упрощения алгоритма Ньютона [6], в котором градиент определяется аналогично LMS-алгоритму, а в качестве корреляционной матрицы  $\mathbf{R}_N$  используется ее регуляризованная оценка

$$\hat{\mathbf{R}}_N(k) = \delta^2 \mathbf{I}_N + \mathbf{x}_N(k) \mathbf{x}_N^H(k),$$

где  $\delta^2 \geq 0.01\sigma_x^2$ . Регуляризация улучшает обращение матрицы  $\mathbf{x}_N(k) \mathbf{x}_N^H(k)$ , которая может быть плохо обусловленной из-за того, что в ее формировании используется только один отсчет вектора входных сигналов линейного сумматора адаптивного фильтра.

Используя лемму об обращении матрицы

$$(\mathbf{C} - \mathbf{B}^H \mathbf{A}^{-1} \mathbf{D})^{-1} = \mathbf{C}^{-1} - \mathbf{C}^{-1} \mathbf{B}^H (\mathbf{D} \mathbf{C}^{-1} \mathbf{B}^H - \mathbf{A})^{-1} \mathbf{B} \mathbf{C}^{-1},$$

где в рассматриваемом случае  $\mathbf{C} = \delta^2 \mathbf{I}_N$ ,  $\mathbf{C}^{-1} = \delta^{-2} \mathbf{I}_N$ ,  $\mathbf{B}^H = \mathbf{x}_N(k)$ ,  $\mathbf{D} = \mathbf{x}_N^H(k)$  и  $\mathbf{A}^{-1} = -1$  уравнение вычисления вектора ВК адаптивного фильтра с помощью NLMS-алгоритма принимает вид

$$\mathbf{h}_N(k) = \mathbf{h}_N(k-1) + \frac{\hat{\mu}}{\mathbf{x}_N^H(k) \mathbf{x}_N(k) + \delta^2} \mathbf{x}_N(k) \alpha^*(k).$$

Здесь  $0 < \hat{\mu} < 2$  – масштабирующий множитель динамического шага сходимости,  $\delta^2$  – небольшое число  $\delta^2 > 0.01\sigma_x^2$ , служащее для предотвращения ситуаций деления на ноль.

В случае действительных ВК

$$\mathbf{h}_N(k) = \mathbf{h}_N(k-1) + \frac{2\hat{\mu}}{\mathbf{x}_N^T(k) \mathbf{x}_N(k) + \delta^2} \mathbf{x}_N(k) \alpha(k),$$

а  $0 < \hat{\mu} < 1$ .

Оценка вычислительной сложности NLMS-алгоритма равна  $3N$  умножениям,  $3N$  сложениям и одному делению на одну итерацию. Эту сложность можно уменьшить до  $2N$  арифметических операций за счет рекурсивного вычисления произведения  $\mathbf{x}_N^H(k) \mathbf{x}_N(k)$ . Действительно, так как вектор  $\mathbf{x}_N(k)$  представляет собой сдвинутые отсчеты входного сигнала, то

$$\begin{aligned} \mathbf{x}_N^H(k) \mathbf{x}_N(k) &= x^*(k)x(k) + \mathbf{x}_N^H(k-1) \mathbf{x}_N(k-1) - \\ &- x^*(k-N)x(k-N). \end{aligned}$$

Если выполнить масштабирование

$$p(k) = \frac{\mathbf{x}_N^H(k) \mathbf{x}_N(k)}{N} \approx \sigma_x^2,$$

то получится другой NLMS-алгоритм, в котором осуществляется нормализация к средней мощности

$$\mathbf{h}_N(k) = \mathbf{h}_N(k-1) + \frac{\tilde{\mu}}{p(k) + \tilde{\delta}^2} \mathbf{x}_N(k) \alpha^*(k),$$

где  $\tilde{\mu} = \hat{\mu}/N$ ,  $\tilde{\delta}^2 = \delta^2/N$ . Вычислительная сложность такого NLMS-алгоритма также равна  $2N$  арифметическим операциям.

Значение средней мощности  $p(k)$  входного сигнала адаптивного фильтра можно оценивать как

$$p(k) = \beta p(k-1) + (1-\beta) x(k)x^*(k)$$

при  $p(0) = 0$ ,  $0 < \beta < 1$  и  $\beta \approx 1$ .

При обработке стационарных сигналов LMS- и NLMS-алгоритмы демонстрируют примерно одинаковое поведение. В случае обработки нестационарных сигналов вектор  $\mathbf{x}_N(k)$  может сильно меняться от итерации к итерации, т.е. может меняться норма этого вектора

$$\|\mathbf{x}_N(k)\|_2 = \sqrt{\mathbf{x}_N^H(k) \mathbf{x}_N(k)},$$

а, значит, могут меняться градиент  $\mathbf{x}_N(k) \alpha^*(k)$  и приращение вектора ВК  $\Delta \mathbf{h}_N(k) = \mu \mathbf{x}_N(k) \alpha^*(k)$  в LMS-алгоритме. В NLMS-алгоритме это приращение определяется как

$$\Delta \mathbf{h}_N(k) = \frac{\hat{\mu}}{\mathbf{x}_N^H(k) \mathbf{x}_N(k) + \delta^2} \mathbf{x}_N(k) \alpha^*(k).$$

Нормировка градиента «сглаживает» его резкие изменения при изменениях компонент вектора  $\mathbf{x}_N(k)$ .

В связи с этим при фиксированных значениях параметров  $\mu$  и  $\hat{\mu}$  поведение NLMS-алгоритма при обработке нестационарных сигналов является более устойчивым, чем поведение LMS-алгоритма.

$$\text{Таким образом, множитель } \mu(k) = \frac{\hat{\mu}}{\mathbf{x}_N^H(k)\mathbf{x}_N(k) + \delta^2}$$

в NLMS-алгоритме представляет собой разновидность переменного шага сходимости, зависящего от времени и значений компонент вектора входных сигналов адаптивного фильтра  $\mathbf{x}_N(k)$ .

Идея переменного шага сходимости широко используется для увеличения скорости градиентных алгоритмов адаптивной фильтрации. В общем виде, уравнение вычисления вектора ВК LMS-алгоритма с переменным шагом сходимости  $\mu(k)$  имеет вид

$$\mathbf{h}_N(k) = \mathbf{h}_N(k-1) + \mu(k)\mathbf{x}_N(k)\alpha^*(k).$$

Существует ряд LMS-алгоритмов, в которых этот шаг выбирается на каждой итерации по эвристическим правилам, что сильно ограничивает области применения таких алгоритмов.

В то же время, можно получить уравнение, с помощью которого  $\mu(k)$  находится путем градиентного поиска [7]

$$\begin{aligned} \mu(k) &= \mu(k-1) - \rho \nabla_{\mu}(k) = \\ &= \mu(k-1) + 2\rho \operatorname{Re} \left\{ \alpha^*(k)\alpha(k-1)\mathbf{x}_N^H(k-1)\mathbf{x}_N(k) \right\}, \end{aligned}$$

где  $\rho \leq 1$  – также шаг сходимости.

В этом уравнении в качестве одномерного градиента  $\nabla_{\mu}(k)$  используется производная квадрата модуля мгновенной ошибки по переменной  $\mu(k)$ .

Аналогичное уравнение для NLMS-алгоритма имеет вид

$$\begin{aligned} \hat{\mu}(k) &= \hat{\mu}(k-1) - \rho \nabla_{\hat{\mu}}(k) = \\ &= \hat{\mu}(k-1) + 2\rho \operatorname{Re} \left\{ \alpha^*(k)\tilde{\alpha}(k-1)\mathbf{x}_N^H(k-1)\mathbf{x}_N(k) \right\}. \end{aligned}$$

Выбор параметра  $\rho$  в этих уравнениях является неоднозначной задачей, поскольку не существует аналитического критерия для определения его предельных значений. Однако известно, что этот параметр должен быть таким, чтобы значения  $\mu(k)$  удовлетворяли условиям устойчивости алгоритмов адаптивной фильтрации.

Так как параметры  $\mu$  и  $\hat{\mu}$  – это положительные действительные числа, не равные нулю, то, исходя из физических соображений, на каждом шаге необходимо ограничивать значения  $\mu(k)$  и  $\hat{\mu}(k)$  как  $\mu_{\min} \leq \mu(k) \leq \mu_{\max}$  и  $\mu_{\min} \leq \hat{\mu}(k) \leq \mu_{\max}$ , где  $\mu_{\min} > 0$

выбирается из соображения минимально-допустимой скорости сходимости алгоритмов, а  $\mu_{\max} > 0$  – из условий устойчивости.

## VI. ЛИНЕЙНО-ОГРАНИЧЕННЫЕ LMS- И NLMS-АЛГОРИТМЫ

Линейно-ограниченная адаптивная фильтрация сигналов нашла широкое применение в алгоритмах управления адаптивными антенными решетками [8]. В таких решетках может отсутствовать требуемый сигнал. При этом ориентация и защита от разрушения основного луча антенной решетки осуществляются за счет линейного ограничения, поддерживающего требуемый уровень ее диаграммы направленности в направлении на источник полезного сигнала.

Линейные ограничения определяются как  $\mathbf{C}_{NJ}^H \mathbf{h}_N(k) = \mathbf{f}_J$ , где  $\mathbf{C}_{NJ}$  – матрица  $J$  ограничений,  $\mathbf{f}_J$  – вектор  $J$  значений ограничиваемого параметра (например, выходного сигнала адаптивного фильтра).

Суть линейно-ограниченной адаптивной фильтрации основана на том, что минимизируемая MSE-функция модифицируется с помощью множителей Лагранжа. В результате, уравнение линейно-ограниченного LMS-алгоритма поиска ВК адаптивного фильтра выражается как

$$\mathbf{h}_N(k) = \mathbf{P}_N \left[ \mathbf{h}_N(k-1) + \mu \mathbf{x}_N(k)\alpha^*(k) \right] + \hat{\mathbf{f}}_N,$$

где  $\mathbf{P}_N = \mathbf{I}_N - \mathbf{C}_{NJ} \left[ \mathbf{C}_{NJ}^H \mathbf{C}_{NJ} \right]^{-1} \mathbf{C}_{NJ}^H$  – проекционная матрица, а  $\hat{\mathbf{f}}_N = \mathbf{C}_{NJ} \left[ \mathbf{C}_{NJ}^H \mathbf{C}_{NJ} \right]^{-1} \mathbf{f}_J$ . Из-за умножения на квадратную матрицу  $\mathbf{P}_N$  арифметическая сложность данного уравнения пропорциональна  $O(N^2)$  арифметическим операциям. Эту сложность можно уменьшить до  $O(N)$  операций, преобразовав данное уравнение к виду

$$\mathbf{h}_N(k) = \mathbf{h}'_N(k) + \mathbf{Q}_{NJ} \left[ \mathbf{f}_J - \mathbf{C}_{NJ}^H \mathbf{h}'_N(k) \right],$$

где

$$\mathbf{h}'_N(k) = \mathbf{h}_N(k-1) + \mu \mathbf{x}_N(k)\alpha^*(k),$$

$$\mathbf{Q}_{NJ} = \mathbf{C}_{NJ} \left[ \mathbf{C}_{NJ}^H \mathbf{C}_{NJ} \right]^{-1}.$$

Аналогичное уравнение для линейно-ограниченного NLMS-алгоритма [6] имеет вид

$$\mathbf{h}'_N(k) = \mathbf{h}_N(k-1) + \hat{\mu} \mathbf{x}_N(k)\tilde{\alpha}^*(k),$$

$$\tilde{\alpha}(k) = \alpha(k) \times$$

$$\times \left[ \mathbf{x}_N^H(k)\mathbf{x}_N(k) + \left[ \mathbf{x}_N^H(k)\mathbf{Q}_{NJ} \right] \left[ \mathbf{C}_{NJ}^H \mathbf{x}_N(k) \right] + \delta^2 \right]^{-1}.$$

Геометрическая интерпретация процедуры вычисления вектора ВК адаптивного фильтра с помощью линейно-ограниченного LMS-алгоритма показана на рис. 5. В результате вычислений вектор

ВК всегда упирается в плоскость  $H_1 = \{\mathbf{h}_N : \mathbf{C}_{NJ}^H \mathbf{h}_N(k) = \mathbf{f}_J\}$ , т. е. удовлетворяет условиям ограничений.

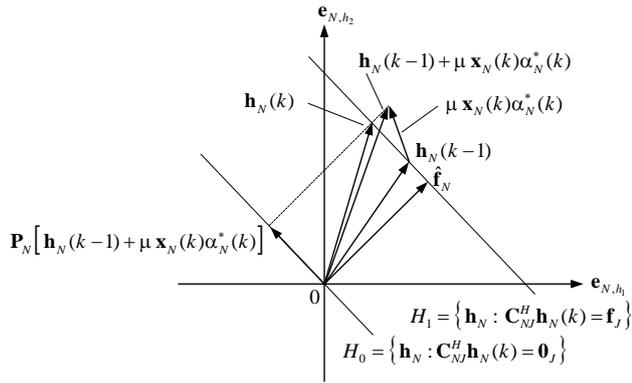


Рис. 5. Вычисление ВК в LMS-алгоритме

Линейно-ограниченный LMS-алгоритм обладает свойством устранения ошибок, накопленных при вычислении вектора ВК. Если в результате этих ошибок вектор  $\mathbf{h}_N(k-1)$  выходит за пределы плоскости ограничений, то на следующей итерации  $k$  процедура проецирования возвращает следующий новый вектор  $\mathbf{h}_N(k)$  на плоскость ограничений, см. рис. 6.

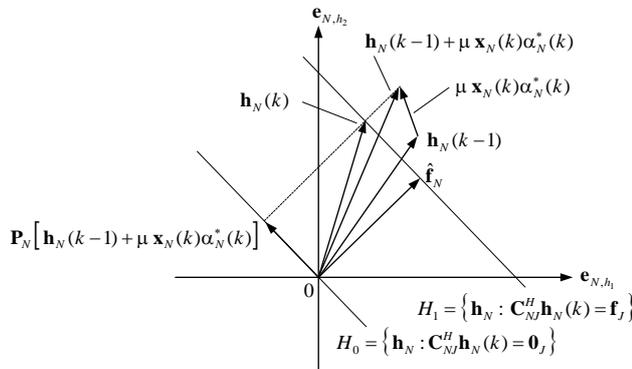


Рис. 6. Коррекция вектора ВК в LMS-алгоритме

## VII. ДРУГИЕ РАЗНОВИДНОСТИ LMS-АЛГОРИТМА

Кроме рассмотренных, существуют версии LMS-алгоритма, в которых в целях упрощения вместо значений обрабатываемых сигналов используются их знаки [9]. Поскольку при этом вычисляется грубая оценка градиента целевой функции, то такие алгоритмы характеризуются более длительным переходным процессом по сравнению с неупрощенными алгоритмами.

Сегодня являются популярными и градиентные алгоритмы на базе субполосной фильтрации обрабатываемых сигналов. Существуют две основные

разновидности таких алгоритмов: на базе процедур быстрого преобразования Фурье (БПФ) и на базе гребенок фильтров [10, 11].

Центральной идеей применения субполосной адаптивной фильтрации является то, что при ее использовании уменьшается разброс собственных чисел корреляционной матрицы в каждой из разделенных частотных полос, что позволяет для обработки в этих полосах использовать отдельные адаптивные фильтры с разными шагами сходимости. В случае обработки сигналов, корреляционная матрица которых характеризуется большим разбросом собственных чисел, субполосные адаптивные фильтры демонстрируют большую скорость сходимости по сравнению с адаптивными фильтрами с обработкой сигналов во всей полосе частот. Субполосные адаптивные фильтры характеризуются пониженной вычислительной сложностью по сравнению с алгоритмами, обрабатывающими сигналы в полной полосе. Это достигается за счет блочной обработки и использования процедур БПФ.

## VIII. СРЕДСТВА МОДЕЛИРОВАНИЯ АДАПТИВНЫХ ФИЛЬТРОВ

Сегодня для исследования адаптивных фильтров широко используется математическое моделирование. Ряд адаптивных фильтров на базе различных алгоритмов, включая LMS-алгоритм и его разновидности, реализован в стандартных библиотеках известных языков программирования и сред моделирования MATLAB [12 – 14], LabVIEW [15] и SystemVue [16]. Кроме того, среда LabVIEW и аппаратные средства компании National Instruments [15] позволяют реализовывать в FPGA адаптивные фильтры на базе LMS-алгоритма (рис. 7).

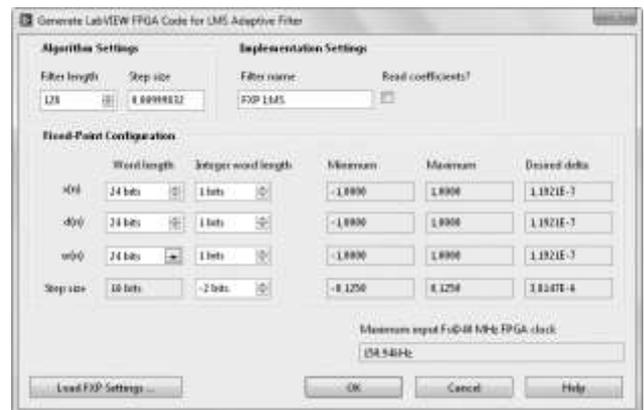


Рис. 7. Целочисленная реализация адаптивного фильтра в FPGA

Пример стандартного виртуального прибора среды проектирования LabVIEW показан на рис. 8, а нестандартного – на рис. 9. Аналогичные приборы могут быть построены и в среде SystemVue [16].

Более детальное описание различных средств моделирования адаптивных фильтров и примеры их реализации на современных ЦСП различных компаний [17 – 20] приведено в [6].

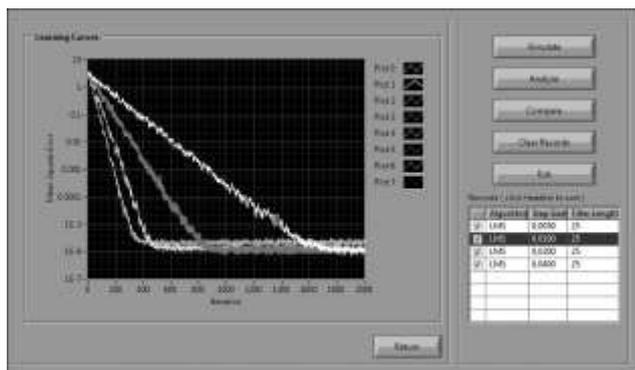


Рис. 8. Демонстрационная версия LMS-алгоритма

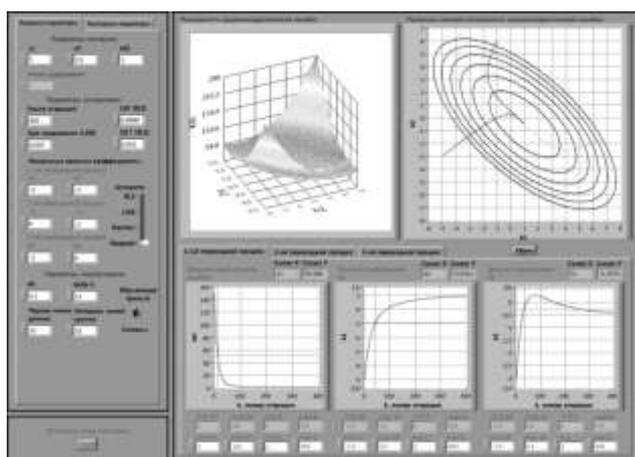


Рис. 9. Виртуальный прибор адаптивного фильтра с двумя ВК

#### IX. ЗАКЛЮЧЕНИЕ

Таким образом, LMS-алгоритм адаптивной фильтрации и его разновидности, благодаря своей низкой вычислительной сложности и простоте, являются, и еще долгие годы будут оставаться «рабочими лошадками» в ЦОС и ее приложениях. Альтернатива этому алгоритму – более сложные, но и более эффективные с точки зрения скорости сходимости, RLS-алгоритмы [21], а также алгоритмы аффинных проекций [22]. Однако эти алгоритмы при аппаратной реализации занимают большую площадь на кристалле СБИС, а при программной реализации для них требуется большая вычислительная мощность по сравнению с LMS-алгоритмом. Программная реализация сегодня уже не является особой проблемой [13]. Недалек и тот час, когда достижения микроэлектронных технологий позволят столь же эффективно реализовать сложные адаптивные алгоритмы и аппаратно.

Пока же адаптивные фильтры на базе LMS-алгоритма используются в большинстве изделий современной радиоэлектронной техники.

#### ЛИТЕРАТУРА

- [1] Джиган В.И. История, теория и практика адаптивной обработки сигналов // Проблемы разработки перспективных микро- и наноэлектронных систем - 2012». Сб. трудов / под общ. ред. академика РАН А.Л. Стемпковского. М.: ИППМ РАН, 2012. С. 30-37.
- [2] Widrow B. Hoff M.E. Adaptive switching circuits // IRE WESCON Convention Record. 1960. Part 4. P. 96–104.
- [3] Widrow B. Thinking about thinking: the discovery of the LMS algorithm – DSP history // IEEE Signal Processing Magazine. 2005. Vol. 22. № 1. P. 100–106.
- [4] Уидроу Б., Стирнз С.Д. Адаптивная обработка сигналов / Пер. с англ. под ред. Шахгильдяна В.В. М.: Радио и связь, 1989. 440 с.
- [5] Farhang-Boroujeny B. Adaptive filters. Theory and applications. 2-nd edition. John Wiley and Sons, 2013. 778 p.
- [6] Джиган В.И. Адаптивная фильтрация сигналов: теория и алгоритмы. М.: Техносфера, 2013. 528 с.
- [7] Mathews V.J., Xie Z. A stochastic gradient adaptive filter with gradient adaptive step size // IEEE Trans. Signal Processing. 1993. Vol. 41. № 6. P. 2075–2087.
- [8] Frost O.L. An algorithm for linearly constrained adaptive array processing // Proc. of the IEEE. 1972. Vol. 60. № 8. P. 926–935.
- [9] Diniz P.S.R. Adaptive filtering algorithms and practical implementation. Fourth edition. Springer Science, 2012. 652 p.
- [10] Lee K.-A., Gan W.-S., Kuo S.M. Subband adaptive filtering: theory and implementation. John Wiley and Sons, Inc. 2009. 344 p.
- [11] Soo J.-S., Pang K. K. Multidelay block frequency domain adaptive filter // IEEE Trans. Signal Processing. 1990. Vol. 38. № 2. P. 373–376.
- [12] Сергиенко А.Б. Цифровая обработка сигналов: учебное пособие. 3-е изд. СПб: БХВ-Петербург, 2011. 768 с.
- [13] Джиган В.И. Прикладная библиотека адаптивных алгоритмов // Электроника: Наука, Технологии, Бизнес. 2006. № 1. С. 60–65.
- [14] <http://www.mathworks.com> // Интернет-сайт компании MathWorks (дата обращения: 02.04.2014).
- [15] <http://www.ni.com> // Интернет-сайт компании National Instruments (дата обращения: 02.04.2014).
- [16] <http://www.agilent.com> // Интернет-сайт компании Agilent Technologies (дата обращения: 02.04.2014).
- [17] <http://www.ti.com> // Интернет-сайт компании Texas Instruments (дата обращения: 02.04.2014).
- [18] <http://www.analog.com> // Интернет-сайт компании Analog Device (дата обращения: 02.04.2014).
- [19] <http://www.freescale.com> // Интернет-сайт компании Freescale Semiconductor (дата обращения: 02.04.2014).
- [20] <http://www.multicore.ru> // Интернет-сайт ОАО НПЦ «ЭЛВИС» (дата обращения: 02.04.2014).
- [21] Джиган В.И. Многоканальные RLS- и быстрые RLS-алгоритмы адаптивной фильтрации // Успехи современной радиоэлектроники. 2004. № 11. С. 48–77.
- [22] Ozeki K., Umeda K. An adaptive filtering algorithm using orthogonal projection to an affine subspace and its properties // Trans. IECE Japan. 1984. Vol. J67A. № 2. P. 126–132.