

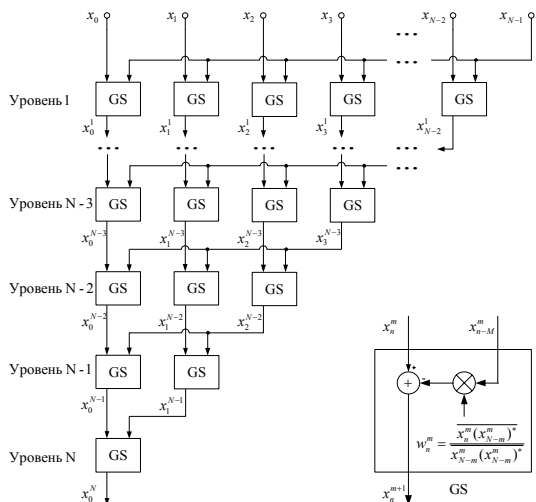
## БЫСТРАЯ ПРОЦЕДУРА ОРТОГОНАЛИЗАЦИИ ГРАМА-ШМИДТА НА ОСНОВЕ АЛГОРИТМА NLMS

Орешкин Б.Н.<sup>1</sup>, Бакулев П.А.<sup>1</sup>, Кирпичников А.П.<sup>2</sup>

<sup>1</sup>Московский Авиационный Институт (Государственный Технический Университет)

<sup>2</sup>Институт Проблем Управления Российской Академии Наук, [info@autex.ru](mailto:info@autex.ru)

**Аннотация** – предложен способ вычисления коэффициентов преобразования классической процедуры ортогонализации Грама-Шмидта (GS), основанный на применении адаптивного алгоритма Normalized Least Mean Squares (NLMS). Представлены структурная схема алгоритма и результаты моделирования, подтверждающие его эффективность. Проведен также анализ вычислительных затрат, необходимых для физической реализации алгоритма.



### I. ПОСТАНОВКА ЗАДАЧИ

Пусть  $x_0(t), x_1(t), \dots, x_{N-1}(t)$  представляют собой комплексные входные векторы процедуры ортогонализации GS в ее классическом виде применительно к задачам ЦОС [3] (

Рис. 1). Математически эта структура эквивалентна следующему алгоритму [2]:

$$x_n^{m+1} = x_n^m - (w_n^m)^* x_{N-m}^m, \quad n = 0, 1, \dots, N - m - 1,$$

$m = 0, 1, \dots, N - 1$ , где \* - операция комплексного сопряжения.

Классическая процедура ортогонализации GS содержит большое количество операций деления, необходимых для вычисления коэффициентов взаимной корреляции сигналов на каждом уровне преобразования. Веса адаптивного алгоритма вычисляются из условия ортогональности входного и выходного векторов процедуры ортогонализации:

Рис. 1.

$$M(x_n^{m+1} (x_{N-m}^m)^*) = 0 \Rightarrow w_n^m = M(x_n^m (x_{N-m}^m)^*) / M(x_{N-m}^m (x_{N-m}^m)^*) = \rho_n^m,$$

где  $M$  – операция вычисления математического ожидания. Такой подход к вычислению весовых коэффициентов справедлив только при обработке стационарных входных сигналов. Если входные сигналы нестационарные, то возникает необходимость подстройки  $w_{n,k}^m$  во времени ( $k$  – номер отсчета). Количество делений, необходимых для реализации алгоритма при обработке каждого отсчета в этом случае равно количеству уровней ортогонализации  $N$ . Значения весовых коэффициентов сглаживаются экспоненциальным фильтром ( $\alpha \in (0, 1)$ ) [4]:

$$r_{n,k}^m(t) = \alpha r_{n,k}^m(t-1) + (1 - \alpha) x_{n,k}^m(t) (x_{N-m,k}^m(t))^*,$$

Рис. 1 Структурная схема процедуры ортогонализации Грама-Шмидта  $r_{N-m,k}^m(t) = \alpha r_{n,k}^m(t-1) + (1 - \alpha) x_{N-m,k}^m(t) (x_{N-m,k}^m(t))^* w_{n,k}^m = r_{n,k}^m / r_{N-m,k}^m$ .

Сформулируем задачу оптимизации следующим образом. *Задача:* минимизировать количество операций деления, необходимых для вычисления весовых коэффициентов процедуры ортогонализации GS.

### II. БЫСТРАЯ ПРОЦЕДУРА ОРТОГОНАЛИЗАЦИИ ГРАМА-ШМИДТА

Для решения поставленной задачи предлагается использование для вычисления значений весовых коэффициентов процедуры ортогонализации GS алгоритма

NLMS. Известно [1], что при адаптации одного весового коэффициента посредством алгоритма NLMS величина этого весового коэффициента сходится в пределе к значению

$$\text{коэффициента взаимной корреляции входных сигналов: } \lim_{t \rightarrow \infty} w_n^m(t) = \rho_n^m.$$

При этом величина шага адаптации  $\gamma_k^m$  нормируется в соответствии с мощностью входного сигнала на каждом уровне ортогонализации:  $\gamma_k^m = \frac{\gamma_0}{M(x_{N-m,k}^m (x_{N-m,k}^m)^*)}$ ,  $\gamma_0 \in (0, 0.25)$ .

Однако, в таком виде предложенный алгоритм (Рис. 2):

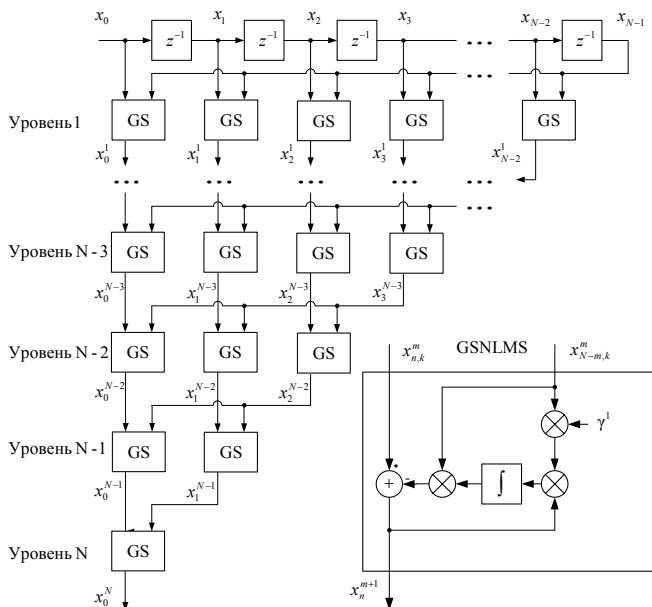
$$w_{n,k}^m(t+1) = w_{n,k}^m(t) + 2\gamma_k^m x_{n,k}^m(t+1) (x_{N-m,k}^m(t+1))^*,$$

все еще требует применения  $N$  операций деления. Эксперименты показывают, что следующая аппроксимация шага адаптации на каждом уровне ортогонализации не приводит к сколько-нибудь

заметному ухудшению характеристик работы алгоритма:

$$\gamma_k^m = \gamma_k^{m-1} = \dots = \gamma_k^2 = \gamma_k^1 = \frac{\gamma_0}{M(x_{N-1,k}(x_{N-1,k})^*)}$$

Структурная схема и алгоритм предложенной процедуры ортогонализации GS на основе алгоритма NLMS (GSNLMS), требующей только одной операции деления для вычисления весовых коэффициентов, приведены на Рис. 2.



Алгоритм GSNLMS :

инициализация:

$$\gamma_0 = 0.01; \quad \gamma_1 = \gamma_0 / P_{\max};$$

вычисления:

for  $k = 1 : K$

$$x_{0:N-1}^0 = 0; \quad R_{N-1}^0 = 0.01;$$

for  $n = 0 : M - 1$

for  $m = 0 : N - 1$

for  $l = 0 : N - 1 - m$

$$x_l^{m+1} = x_{l+1}^m - (w_l^m)^* \cdot x_{N-m}^m;$$

$$w_l^m = w_l^m + (x_l^{m+1})^* \cdot temp;$$

end;

$$temp = \gamma^1 \cdot x_{N-m}^m;$$

end;

$$R_{N-1}^0 = R_{N-1}^0 + x_{N-1} \cdot (x_{N-1})^*;$$

end;

$$\gamma^1 = 2 \cdot K \cdot \gamma^0 / R_{N-1}^0;$$

end;

Рис. 2 Структурная схема, базисный элемент и алгоритм процедуры ортогонализации GSNLMS

### III. РЕЗУЛЬТАТЫ МОДЕЛИРОВАНИЯ И АНАЛИЗ ВЫЧИСЛИТЕЛЬНЫХ ЗАТРАТ

Результаты компьютерного моделирования и анализ вычислительных затрат алгоритмов GS, GSNLMS и Recursive GS (RGS [4]) показаны на Рис. 3 и в Таблице 1 соответственно.

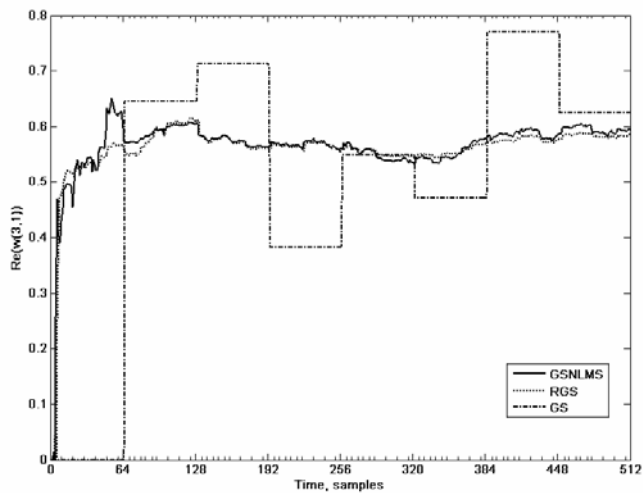


Таблица 1

+  
×  
÷

GS  
 $N^2 + N$   
 $N^2 + N$   
 $N$

RGS  
 $N^2$   
 $2N^2 + N$   
 $N$

GSNLMS  
 $N^2 + 1$   
 $N^2 + N + 1$   
1

Рис. 3 Процесс адаптации коэффициента  $w_1^3$  для алгоритмов GS, RGS и GSNLMS

Из рисунка и таблицы видно, что предложенный алгоритм GSNLMS не только обеспечивает значительную экономию вычислительных ресурсов, но и не уступает алгоритму RGS по скорости сходимости и точности настройки весовых коэффициентов.

#### Литература

1. S. Haykin, Adaptive filter theory, 3<sup>rd</sup> ed. Englewood Cliffs, NJ: Prentice-Hall, 1996.
2. K. Gerlach, F. F. Kretschmer Jr, Convergence Properties of Gram-Schmidt and SMI Adaptive algorithms, IEEE Transactions on Aerospace and Electronic Systems, Vol. 26, No. 1, January 1990.
3. K. Gerlach, Convergence Bounds of an SMI/Gram-Schmidt Canceller in Colored Noise, IEEE Transactions on Aerospace and Electronic Systems, Vol. 24, No. 4, July 1991.
4. X. Huang, M. Caron, D. Hindson, A Recursive Gram-Schmidt Orthonormalization Procedure and its Application to Communications, Third IEEE Signal Processing Workshop on Signal Processing Advances in Wireless Communications, Taoyuan, Taiwan, March 20-23, 2001.

