Journal of Applied Sciences

ISSN 1812-5654

science alert

ANSInet
an open access publisher
http://ansinet.com

Journal of Applied Sciences 9 (4): 759-764, 2009
ISSN 1812-5654

Combining Several PBS-LMS Filters as a General Form of Convex Combination of Two Filters

A. Fathiyan and M. Eshghi
Department of Electrical and Computer Engineering, Shahid Beheshti University,
P.O. Box 14455-463, Tehran, Iran

Abstract: Combination approaches can improve the performance of adaptive filters. Recently a convex combination of adaptive filters was proposed to improve the performance of LMS algorithm. Our proposal in this study is to use the PBS-LMS algorithm instead of LMS algorithm in the structure of convex combination. Our simulations showed that this structure not only has the optimality of first one, but also, it has the features of PBS-LMS algorithm such as regularity. By using PBS-LMS algorithm in this structure we saved in total number of samples needed by filter to converge about 22.2%, for example the fast filter converged to the steady state in 352 samples, the slow one in 397 samples and the overall filter in 309 samples. Also, this scheme was generalized, combining multiple PBS-LMS filters with different adaptation step sizes.

Key words: Adaptive filters, convex combination, PBS-LMS algorithm

INTRODUCTION

The Least Mean Square (LMS) algorithm has been extensively used in many applications due to its simplicity and robustness (Haykin, 2002). There are two main difficulties concerning LMS algorithm. The first arises in situations with a high eigenvalue spread in the correlation matrix of the input process. The second difficulty is the inherent balance between speed of convergence and final misadjustment in stationary situations that is imposed by the selection of a certain value for the adaptation step size (Arenas-García *et al*., 2003).

To solve the second problem some previous articles try to improve the speed vs precision balance by using non-quadratic error functions [e.g. the Least Mean Fourth (LMF) algorithm (Walach and Widrow, 1984)] that get a faster convergence. Martínez-Ramón *et al.* (2002) combined one fast and one slow LMS filters with the objective of getting the advantages of both of them: Fast convergence and good tracking capabilities from the fast LMS filter and reduced steady-state error from the slow one. The mean-square performance of this structure have been studied by Arenas-García *et al*. (2006). In particular, they showed that the combination filter structure is universal (Singer and Feder, 1999; Merhav and Feder, 1998) in the sense that it performs, in the mean-square error sense, as well as the best of its components. Arenas-García *et al*. (2003) proposed to use multiple filters in this structure which is the natural extension of the convex combination of two LMS filters (CLMS) of Martínez-Ramón *et al*. (2002). Zhang and Chambers (2006) used this combination for the first time together with the fractional tap-length method to solve the optimal filter tap-length search problem in a high noise environment, where, SNR≤0 dB.

Our proposal in this study is to use PBS-LMS algorithm (Eshghi and DeGroat, 1995) instead of LMS algorithm in the structure of this convex combination. PBS-LMS algorithm (parallel binary structured-LMS) is a parallel algorithm for implementing adaptive filters. The basic advantage of this algorithm over other parallel algorithms is that in those algorithms coefficient vector is updated at any time step, but in PBS-LMS algorithm the coefficient vector is updated after s time steps. Therefore by applying this algorithm the number of calculations for updating coefficient vector decreases significantly and speedup increases (Majdar and Eshghi, 2004).

MATERIALS AND METHODS

Convex combination of adaptive filters: The CLMS filter (Martínez-Ramón *et al*., 2002) uses a convex combination of the weights of the two LMS filters as shown in Fig. 1. The output signals and the output errors of both filters are combined in such a way that the advantages of both component filters are retained: the rapid convergence from the fast filter and the reduced steady-state error from the slow filter. The output of the overall filter is:

Corresponding Author: Ali Fathiyan, Department of Electrical and Computer Engineering, Shahid Beheshti University, P.O. Box 14455-463, Tehran, Iran Tel: +989127119801 Fax: +982122417940

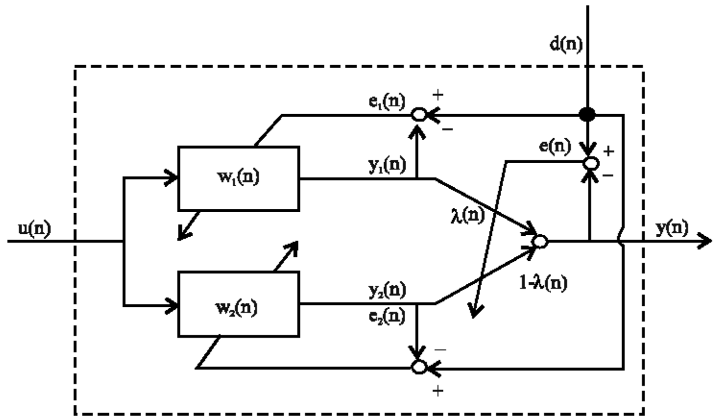

Fig. 1: The structure of convex combination of two LMS filters

$$y(n) = \lambda(n)y_1(n) + (1-\lambda(n))y_2(n) \quad (1)$$

where, $y_i(n) = w_i^T(n)\times(n)$ and $w_i(n)$, $x(n)$ and $\lambda(n)$ are the adaptive filter weight vector, input vector and mixing scalar parameter for i = 1, 2 respectively. The idea is that if $\lambda(n)$ is assigned appropriate values at each iteration, then the above combination would extract the best properties of the individual filters $w_1(n)$ and $w_2(n)$. Both filters operate completely decoupled from each other, using standard LMS adaptation rules:

$$w_i(n+1) = w_i(n) + \mu_i e_i(n)\times(n) \quad (2)$$

where, $e_i(n)$ is the error produced by each filter at time step n, i.e., $e_i(n) = d(n) - w_i^T(n)\times(n)$ and d(n) is the desired output.

The CLMS filter uses a convex combination of the weights of the two LMS filters:

$$w_{eq}(n) = \lambda(n)w_1(n) + (1-\lambda(n))w_2(n) \quad (3)$$

where, parameter $\lambda(n)$ is kept in interval [0, 1] by defining it as:

$$\lambda(n) = \mathrm{sgm}(a(n)) = 1/(1+e^{a(n)}) \quad (4)$$

The combination parameter is adapted to minimize the error of the overall adaptive filter, also using LMS adaptation rule:

$$a(n+1) = a(n) + \frac{\mu_a \partial e_{eq}^2(n)}{2\partial a(n)} = a(n) + \mu_a(d(n) - w_{eq}^T(n)\times(n)) \quad (5)$$
$$(\lambda(n)(1-\lambda(n))(w_1(n) - w_2(n))^T\times(n)$$

In the above equation, μ_a must be fixed to a value much higher than μ_1 so, the combination is adapted even faster than the fastest of the LMS filters. Note that the update of a(n) in Eq. 5 stops whenever $\lambda(n)$ is too close to the limit value of zero or one. To overcome this problem, we shall restrict the values of a(n) to lie inside a symmetric interval $[-a^+, a^+]$, which limits the permissible range of $\lambda(n)$ to $[1-\lambda^+, \lambda^+]$, where, $\lambda^+ = \mathrm{sgm}(a^+)$ is a constant close to one. In this way, a minimum level of adaptation is always guaranteed.

This scheme has a very simple interpretation: in situations where a high speed would be desirable, the fast LMS will outperform the slow one, making $\lambda(n)$ approaches towards 1 and $w_{eq}(n)\approx w_1(n)$. However, in stationary intervals, it is the slow filter which operates better, making $\lambda(n)$ get close to 0 and $w_{eq}(n)\approx w_2(n)$.

It is possible to further improve the performance of the basic combination algorithm by using the good convergence properties of the fast filter to speed up the convergence of the slow LMS filter. Arenas-García *et al.* (2003) did this by step-by-step transferring a part of weight vector w_1 to w_2. So, in this case, the adaptation rule for the slow filter becomes:

$$w_2(n+1) = a(w_2(n) + \mu_2 e_2(n)\times(n) + (1-a)w_1(n+1) \quad (6)$$

The weight transfer must only be applied when the fast LMS is performing significantly better than the slow one.

Although the CLMS algorithm requires the introduction of some extra parameters, we will see that their selection is very easy and is not critical and optimal values are not very dependent on the particular concrete scenario in which the filter is being applied by Arenas-García *et al.* (2003).

PBS-LMS algorithm: Eshghi and DeGroat (1995) describe the PBS-LMS algorithm. This algorithm is a parallel form

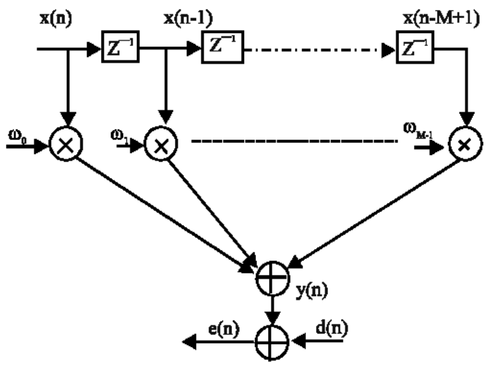

Fig. 2: Transversal adaptive filter

of the LMS algorithm. In this method instead of updating weight vector at any iteration, we reformulate the updating formula of the LMS algorithm in such a way that we will be able to calculate the weight vector at time n+s with respect to the weight vector at time n. Now suppose that there is a transversal filter of order M, as depicted in Fig. 2. The output of this filter at time n is:

$$y(n) = x^H(n)w(n) \tag{7}$$

If d(n) denotes the desired output at time n, then the error at this time is defined as:

$$e(n) = d(n) - y(n) \tag{8}$$

The difference vector for s step update, Δw(n+s), is defined as:

$$\Delta w(n+s) = w(n+s) - w(n) \tag{9}$$

The look-ahead error p is defined as:

$$p(n+k) = x^H(n+k)w(n) - d^*(n+k) \quad k \ge 0 \tag{10}$$

The weighted input vector $\hat{x}(n+k)$ is defined as:

$$\hat{x}(n+k) = -\mu x(n+k) \quad k \ge 0 \tag{11}$$

where, B_r is an ordered set of integer numbers which indicate the positions of 1's in the binary presentation of positive number r.

The scalar c is defined as:

$$c(i,j) = x^H(n+i)\hat{x}(n+j) \qquad i = 1,\dots,s-1 \\ j = 1,\dots,i-1 \tag{12}$$

For any positive number r, c_r is the set of all pairs of adjacent elements of B_r when it has more than one element, i.e., $\|B_r\| > 1$. If b_r has only one element, i.e., $\|B_r\| = 1$, then c_r is NULL.

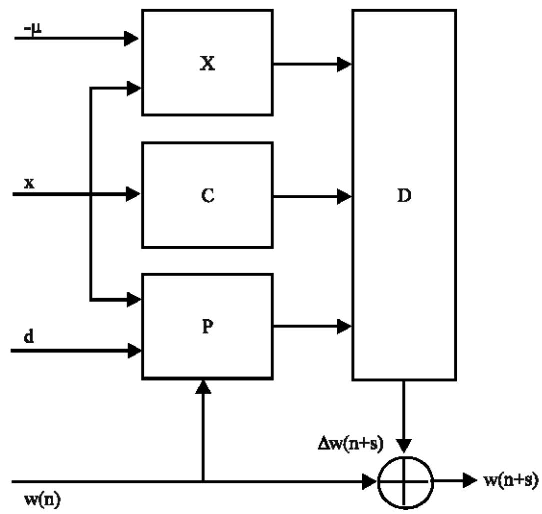

Fig. 3: The block diagram of PBS-LMS algorithm

The PBS-LMS algorithm is as follows:

- Let a_r and b_r be the smallest and the largest elements of B_r, respectively, i.e., $a_r = \min(B_r)$ and $b_r = \max(B_r)$
- For all numbers r with s bits:
  - Assign $p(n+a_r)$ for the right most 1 in the number at a_r^{th} position
  - Assign $\hat{x}(n+b_r)$ for he left most 1 in the number at position
  - Calculate vector v_r:

$$v_r = \begin{cases} \hat{x}(n+a_r)p(n+a_r) & \text{if } r = 2^{a_r} \\ \hat{x}(n+b_r)\left[\prod_{(i,j)\in c_r} c(i,j)\right]p(n+a_r) & \text{O.W.} \end{cases} \tag{13}$$

- Add all v_r's from r = 1 to r = $2^s - 1$ and obtain the difference vector, Δw(n+s) as:

$$\Delta w(n+s) = \sum_{r=1}^{2^s-1} v(r) \tag{14}$$

- Then calculate the weight vector for s step update, i.e., w(n+s), as:

$$w(n+s) = w(n) + \Delta w(n+s) \tag{15}$$

The block diagram of the algorithm is shown in Fig. 3.

The calculations that must be performed in each block are as follows:

In the X block the $\hat{x}(n+i)$ terms are calculated as:

$$\hat{x}(n+i) = -\mu x(n+i) \quad i = 1,\ldots, s-1 \tag{16}$$

In the C block the c coefficients are computed as:

$$c(i,j) = x^H(n+i)\hat{x}(n+j) \quad i = 1,\ldots, s-1 \quad j=1,\ldots,i-1 \tag{17}$$

In the P block p's are calculated as:

$$p(n+k) = x^H(n+k)w(n) - d^*(n+k) \quad k \geq 0 \tag{18}$$

In the D block v_r's are computed and summed to produce.

RESULTS AND DISCUSSION

Present proposal is to use the PBS-LMS algorithm in the convex combination structure to utilize the properties of this algorithm. The proposed algorithm has a parallel structure in two senses. One is the sense that it updates the weight vector for s step ahead. The second sense is the effect of two filters that are working in parallel. The structure of proposed scheme is shown in Fig. 4.

To show the effectiveness of our algorithm, we carried out computer simulation. Two filters are used with different step sizes, one with large step size and the other with a small step size. As the input for both filters, we use a random process with different eigenvalue spreads.

To produce this random process, a random variable u(k) is applied to an Auto Regressive (AR) filter. u(k) is a Gaussian white noise random variable with zero mean and variance of σ_u. The AR filter has a transfer function (Eshghi and DeGroat, 1995) in the form of:

$$H(z) = \frac{1}{1 + az^{-1} + bz^{-1}} \tag{19}$$

The output of the AR filter is a zero mean random process (RP), x(k). The variance of the random process, σ_x, is $\sigma_x = \rho\sigma_x$ where:

$$\rho = ((1+b)^2 - a^2)\,\frac{1+b}{1-b} \tag{20}$$

In order to have zero mean, unity variance RP, the variance of the white noise u(k) is selected to be:

$$\sigma_u = \frac{1}{\rho} \tag{21}$$

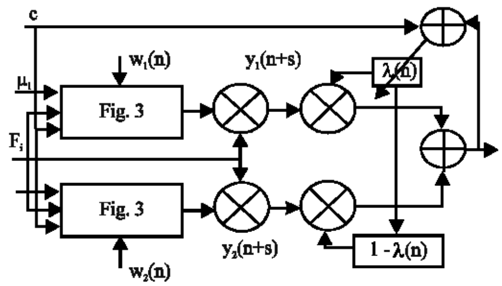

Fig. 4: The structure of proposed scheme

The 2×2 correlation matrix of the above filter has two eigenvalues. These eigenvalues are:

$$\lambda_{max} = \left(1 + \frac{a}{1+b}\right)\sigma_x^2 \tag{22}$$

and

$$\lambda_{min} = \left(1 - \frac{a}{1+b}\right)\sigma_x^2 \tag{23}$$

The eigenvalue spread of the correlation matrix is defined as:

$$\Lambda = \frac{\lambda_{max}}{\lambda_{min}} \tag{24}$$

which is a function of filter variables a and b. By choosing a constant value for a, b is calculated in order to have a defined eigenvalue spread. In our experiment, we select the large step size μ_l = 0.02, the small step size μ_s = 0.005, the step size of mixing scalar parameter μ_a = 20, the tap weight vector of PMS-LMS of order of M = 8 and the look-ahead s = 4. The eigenvalue spread is considered to be Λ = 10. The MSE (Mean Square Error) is shown in Fig. 5.

Figure 5a, depicts the MSE of the slow filter, Fig. 5b shows the MSE of the fast filter and Fig. 5c the MSE of the convex combination. The results show that the error of the slow filter is less than the fast one. On the other hand the fast filter converges to the steady state faster than the slow one. The overall filter, i.e., the convex combination of two filters, has the features of both filters; it has less error than the fast filter and converges to the steady state faster than the slow one. For example to reach an MSD (mean square distortion) of order of 0.001, with a random process of eigenvalue spread of 10, 426 samples is needed in slow filter, 368 samples in fast one and 317 samples in the overall filter, hence saving 25.6% of samples. The results are summarized in Table 1 and 2.

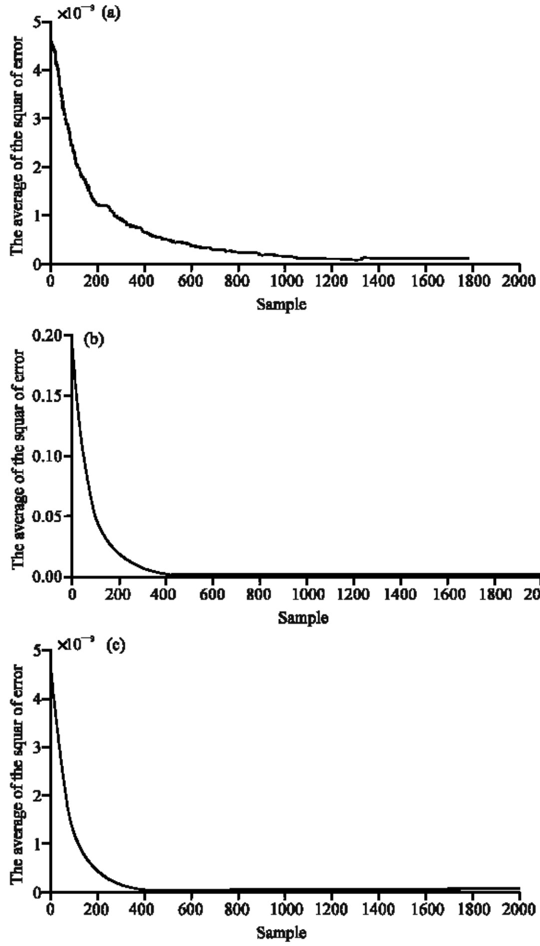

Fig. 5: The error of (a) the slow filter (b) the fast filter and (c) the convex combination of two filters

Combining several PBS-LMS filters: The main topic of this study is the extension of the C-PBS-LMS (convex combination of two PBS-LMS adaptive filters) algorithm to allow the combination of an arbitrary number of individual filters as depicted in Fig. 6. When doing so, the weight vector of the combined M-C-PBS-LMS algorithm becomes:

$$w_{eq}(n) = \sum_{i=1}^{L} \lambda_i(n) w_i(n) \tag{25}$$

L being the total number of PBS-LMS filters that are placed into the combination and $w_i$the weight vector of the i-th PBS-LMS filter with μ_i adaption step size (as before, $\mu_1>\mu_2>...>\mu_L$). As we previously explained for the CLMS algorithm, the L component filters operate, in principle, completely decoupled and adapted using standard LMS rule.

Table 1: Number of samples for convergence of MSE=0.001 in different adaptive filters

	Eigenvalue spread			
Different adaptive filters	2	5	10	20
CLMS (Martínez-Ramón *et al*., 2002)	427	411	368	352
PBS-LMS (Eshghi *et al*., 1995)	512	487	426	397
Proposed scheme in this study	392	372	317	309

Table 2: Percentage of improvement of the proposed scheme over other algorithms

	Eigenvalue spread (%)			
Different adaptive filters	2	5	10	20
CLMS (Martínez-Ramón *et al*., 2002)	8.2	9.5	13.9	12.2
PBS-LMS (Eshghi *et al*., 1995)	23.4	23.6	25.6	22.2

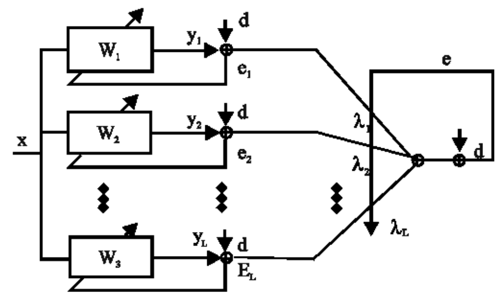

Fig. 6: Combining several PBS-LMS filters

During the derivation of the CLMS algorithm, we were able to see the importance of using a convex combination and limiting the values of λ(n) for the good performance of the algorithm. Similarly, in this case we will use a softmax activation function to obtain the weights assigned to each individual filter:

$$\lambda_i(n) = \frac{\exp(a_i(n))}{\sum_{j=1}^{L} \exp(a_j(n))} \qquad i = 1, 2, \ldots, L \tag{26}$$

which guarantees that $0 \le \lambda_i(n) \le 1$ and $\sum_{i=1}^{L} \lambda_i(n)=1$. The adaption rule for the mixing parameters is:

$$a_i(n+1) = a_i(n) + \mu(d(n) - w_{eq}^T(n) \times (n))\, \lambda_i(n)(w_i(n) - w_{eq}(n))^T \times (n) \tag{27}$$

CONCLUSION

In this study, an approach is presented to use the convex structure to solve the problem of inherent trade-off between convergence speed and misadjustment in the PBS-LMS algorithm (Eshghi and DeGroat, 1995) and also it generalized to use multiple filters in this structure rather than using two filters. Simulation results were given to support the effectiveness of this method.

ACKNOWLEDGMENT

This study was supported by Iran Telecommunication Research Center (ITRC).

REFERENCES

Arenas-García, J., M. Martínez-Ramón, V. Gómez-Verdejo and A.R. Figueiras-Vidal, 2003. Multiple plant identifier via adaptive LMS convex combination. Proceeding of the IEEE International Symposium Intelligent Signal Processing, September 4-6, Budapest, Hungary, pp: 137-142.

Arenas-García, J., A.R. Figueiras-Vidal and A.H. Sayed, 2006. Mean-square performance of a convex combination of two adaptive filters. IEEE Trans. Signal Process., 54: 1078-1090.

Eshghi, M. and J. DeGroat, 1995. A parallel binary structured LMS algorithm for transversal adaptive filters. The J. VLSI Signal Process., 10: 127-140.

Haykin, S., 2002. Adaptive Filter Theory. 4th Edn., Prentice-Hall, New Jersey, ISBN: 0130901261.

Majdar, R.S. and M. Eshghi, 2004. Implementation of PBS/spl I.bar/LMS algorithm for adaptive filters on FPGA. Proceeding of the IEEE Tencon Conference, November 21-24, IEEE Xplore, London, pp: 9-12.

Martínez-Ramón, M., J. Arenas-García, A. Navia-Vázquez and A.R. Figueiras-Vidal, 2002. An adaptive combination of adaptive filters for plant identification. Proceeding of the 14th International Conference on Digital Signal Processing, July 1-3, Santorini, Greece, pp: 1195-1198.

Merhav, N. and M. Feder, 1998. Universal prediction. IEEE Trans. Inform. Theory, 44: 2124-2147.

Singer, A.C. and M. Feder, 1999. Universal linear prediction by model order weighting. IEEE Trans. Signal Process., 47: 2685-2700.

Walach, E. and B. Widrow, 1984. The least mean fourth (LMF) algorithm and its family. IEEE Trans. Inform. Theory, IT-30: 275-283.

Zhang, Y. and J. A. Chambers, 2006. Convex combination of adaptive filters for a variable tap-length LMS algorithm. IEEE Signal Process. Lett., 13: 628-631.