AN EFFICIENT CHANNEL EQUALIZATION TECHNIQUE WITH A LEAKY FLMS ALGORITHM

G. Bhanu Chandra and Abhijit Mitra
Department of Electronics and Communication Engineering,
Indian Institute of Technology (IIT) Guwahati, India.
E-mail: (guduru, a.mitra)@iitg.ernet.in.

Abstract: *We propose a modified version of frequency domain least mean square algorithm, or, fast least mean square (FLMS) algorithm that achieves better convergence in frequency domain with a marginal increment in implementation Complexity than the former one. The proposed algorithm, leaky FLMS (LFLMS), mainly differs from FLMS by virtue of a leaky factor which is dependent on the algorithm step size parameter and improves the convergence behaviour. We apply the proposed algorithm in case of an adaptive channel equalizer. The performance of the proposed algorithm is examined in the said application with respect to the mean square error (MSE) performance as well as bit error rate (BER) versus signal to noise ratio (SNR) curves.*

Keywords: *Least mean square algorithm, Frequency domain, Leaky factor, Bit error rate, Mean square error.*

1. INTRODUCTION

The time-domain least mean square (LMS) adaptive filter algorithm [1] has found many applications in situations where the statistics of the input processes are unknown or changing. These include noise cancelling, line enhancing, and adaptive array processing [2]. The algorithm uses a transversal filter structure driven by a primary input. The filter weights are updated iteratively based upon the difference between the filter output and a reference input, so as to minimize the mean-square error of the difference. In all cases, the stability, convergence time, and fluctuations of the adaptation process are governed by the product of a feedback coefficient and the input sequence power to the adaptive filter. As a result, in all practical applications, there is an implicit automatic gain control (AGC) on the input to the adaptive filter. The AGC ensures that the power-feedback product is maintained within acceptable design limits. When the adaptive filter is implemented as a tapped delay line operating on the entire available input signal bandwidth, selection of a single value of feedback coefficient is required. Then, the algorithm convergence time and stability depends upon the ratio of the largest to the smallest Eigen values associated with the correlation matrix of the input sequence. To avoid this problem, a faster variant of LMS, normalized LMS (NLMS) algorithm, has been proposed with time dependent step size parameter [3].More recently, the computational efficiencies resulting from processing blocks of data, such as the Fast Fourier Transform (FFT) and block digital filtering, has led to the implementation of the LMS adaptive algorithm in the frequency domain. A specific frequency domain implementation of the algorithm was suggested in [4] that promised a significant reduction in computation when the number of weights equalled or exceeded 16. A serial implementation of the time-domain block LMS adaptive filter, that used the frequency domain FFT when implementing the filters, was presented in [5]. Later, for frequency domain applications, a family of fast LMS (FLMS) algorithms [6], [7] have been proposed which can be used in case of large memory applications. They significantly reduce the processing time of the data received. However, it is seen that if somehow the step size approaches the maximum limit, the Algorithm's performance deteriorates very fast if proper care is not taken. Based on this criterion, a leaky factor, dependent on the algorithm step size parameter, is introduced into the algorithm to take care of such conditions. This factor causes the filter to maintain a constant value once large step size is encountered at all frequencies. It is shown that the performance improvement is possible with only a marginal increment in computational complexity. This paper is organized as follows. In Section 2 we deal with fundamentals of FLMS algorithm.

Section 3 introduces the proposed leaky FLMS algorithm and along with its relative computational complexity with respect to FLMS. Experimental results are provided in Section 4. This paper is concluded by summarizing the present t work in Section 5.

2. FUNDAMENTALS OF FLMS ALGORITHM

FLMS algorithm comes under the category of frequency domain adaptive filtering (FDAF) algorithms [8]. There are two main reasons for seeking the use of frequency-domain adaptive filtering in one form or the other. The first one is that in certain applications, such as acoustic echo cancelation in teleconferencing, the adaptive filter is required to have a long impulse response (i.e., long memory) to cope with equally long echo duration. When conventional time domain adaptive signal processing algorithms are used, it results in requirement of a long memory and increase in computational complexity of the algorithm used. The other reason is that the algorithm attains a uniform convergence rate by exploiting the orthogonal properties of discrete Fourier transform (DFT) and related discrete transforms. The various notations and definitions in the context of the FDAF algorithms, to be used in the sequel, are given below.

F[x] = FFT[x],

$F^{-1}[x]$= IFFT[x],

U(k) = diag{F [u(n - N), ...u(n - 1), u(n), ...u(n + N - 1)]}: tap input vector,

W(k)=F[w(n), $\underbrace{0,........0}_{\text{N zeroes}}$]: filter weight vector,

d(n): desired response vector,

y(n): output response vector,

E(k): error vector in frequency domain,

$0_{N\times N}$: N × N all zero matrix,

$I_{N\times N}$: N × N identity matrix

μ: the constant learning rate scalar with the convergence range $0 < \mu < \frac{2}{\mathrm{tr}(R)}$ and

R = E[u T(n) u(n)]: input correlation matrix.

In FLMS algorithm, the desired response vector, output response vector and the error vector in frequency domain are taken as

$$d(n)=[\underbrace{0,.........,0}_{\text{N zeroes}},d(n),....,d(n+N-1)] \quad (1)$$

$$y(n)=\begin{pmatrix} 0_{N\times N} 0_{N\times N} \\ 0_{N\times N} I_{N\times N} \end{pmatrix} F^{-1}[U(K)W(K-1)] \quad (2)$$

and E(k) = F[d(n)-y(n)] (3)

respectively. With the help of (1)-(3), the weight update equation in frequency domain is specified as

$$W(K)=W(K-1)+\mu F\left[\begin{pmatrix} I_{N\times N} O_{N\times N} \\ O_{N\times N} O_{N\times N} \end{pmatrix} F^{-1}[U(K)E(K)]\right] \quad (4)$$

,where the latter half of the right hand side of (4) represents the update factor in frequency domain.

3. THE PROPOSED LEAKY FLMS ALGOR ITHM

In the proposed leaky FLMS (LFLMS) algorithm, except the weight update equation, all the other notations and Definitions remain same. In the weight update equation of FLMS algorithm (4), we introduce a leaky factor (.) to improve the performance of the algorithm. The modified equation, takes the form as

$$W(K) = (1-\mu\gamma)W(K-1) + \mu F \times \left[\begin{pmatrix} I_{N\times N} O_{N\times N} \\ O_{N\times N} O_{N\times N} \end{pmatrix} F^{-1}[U(K)E(K)]\right]$$

Where, the experimental lower and upper limits of are found to be

$$0 \le \gamma \le \frac{2}{\mu} \quad (6)$$

By introducing, we marginally increase the computational complexity as well as the mean square error (MSE) of the algorithm. It is seen that the leaky factor has almost no effect on the small step sizes. However, for large step sizes $\left(\approx \frac{2}{tr(R)}\right)$, the leakage factor prevents the algorithm from diverging and restricts it to maintain a constant Value by putting a certain weightage on the previous filter weight vector.

3.1 A comment on computational complexity

The computational complexities of the algorithms mentioned above namely NLMS, FLMS and Leaky FLMS algorithm is provided in Table 1. Note that, in the table, N denotes the order of the filter. As an example, if we Take FLMS algorithm, there will be 4N additions, 9 log2 N + 6 multiplications and the remaining are for divisions And sorting. Equation (2) takes log2 N for one IFFT operation. In the same way, (3) also takes log2 N for another FFT operation. The remaining is included in (4). The additions can also be explained in the similar way. Before entering the filter the signal is stored in N element block for further usage in the processing. This block serves as the storage block for passing the signal. This explains the storage requirement. We enlist all these in Table 1. It should be noted that for all the algorithms given in the table, the storage requirement is same.

TABLE 1 Computational complexities of LMS, FLMS and LFLMS algorithms

Algorithms	Computational Complexity		Storage Requirement
	Multiplication	Addition	
NLMS	3N+1	3N	N
FLMS	$9\log_2 N + 6$	4N	N
Leaky FLMS	$10\log_2 N + 8$	5N	N

3.2 A few remarks about the LFLMS convergence

Certain mathematical observations have been made regarding the convergence of the proposed leaky FLMS (LFLMS) algorithm. We state them below, without the complicated mathematical proof, as remarks.

Remark 1: First, it must be proved that LFLMS converges. A proof of convergence of FLMS is given in [5] In LFLMS, when $\gamma \to 0$, we observe that LFLMS FLMS, which ensures for small, LFLMS converges.

For $\lim \gamma \to \frac{1}{\mu}$ we observe that E[(1-μ.)W(k)] =0, meaning we are left with only the correction term, which is also a well known term that converges.

Remark 2: It is known that the FLMS algorithm converges in mean to the same solution as that of the LMS.

However, μ must be reduced by N times in FLMS to guarantee stability [5]. When . . 0, we observe that LFLMS, FLMS, and it is the mode we commonly prefer for very small μ values. However, lim..1/μ E[(1 - *μ.)W(k)] = 0,* $\forall W(k) \neq 0$*, for which the optimum value of* γ *. is preferred to be* $\frac{1}{\mu}$

Remark 3: For 0 < μ < $\frac{2}{\lambda_{\max}}$ (where $\lambda_{\max}$ is the largest eigenvalue of R), it is seen that in FLMS algorithm, the misadjustment is reduced by N times to that of the LMS algorithm, i.e., $MSD_{FLMS} = \frac{\mu}{2N} tr(R)$.

The Adaptation accuracy, or, the misadjustment, in LFMLS is the same as that of FLMS which mainly comes from the correction term.

4. RESULTS AND DISCUSSIONS

We generate a Gaussian random variable, to be used as the input, with variance 0.01 and the filter length N is taken as 10. With this, the upper limit of μ comes out to be 2. We have worked with μ = 0.2 and μ = 2 in order to show the effectiveness of the proposed LFLMS algorithm. The channel impulse response used here is h(n) = [0.7 , 0.6, 0.1, 0.1, 0.1, 0.1, 0.1, 0.1, 0.1, 0.2]. The algorithm is tested on 5000 test data, averaged on 100 independent trials.

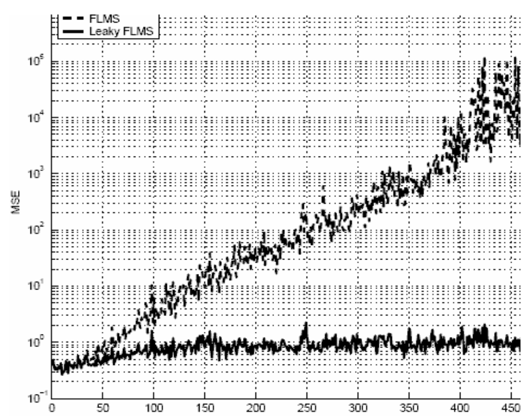

Fig. 1 MSE plots for FLMS and LFLMS algorithms with μ = 2 and γ = 0.5

Fig. 1 shows the MSE plots of FLMS and LFLMS algorithms with μ = 2. Here, γ is taken as 0.5 (μ) as said earlier and both the MSE characteristics are observed. From the figure we can conclude that leaky factor settles down the algorithm's error value to a constant one.

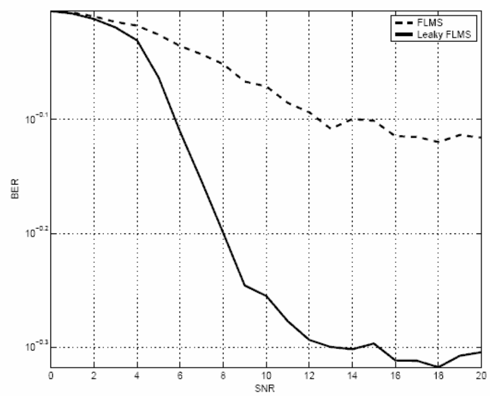

Fig. 2 BER versus SNR plots for FLMS and LFLMS algorithms with $\mu = 2$ and $\gamma = 0.5$

Fig. 2 shows the BER versus SNR plots of FLMS and LFLMS algorithms with the parameters as said earlier. Here we can see that the error rate is much more high for FLMS algorithm than the LFLMS at high SNR values.

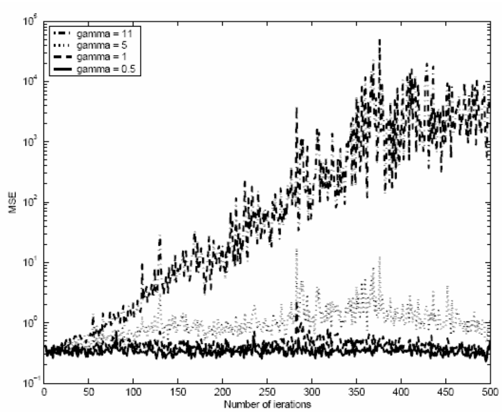

Fig. 3 MSE plots for LFLMS algorithm with dfferent γ values

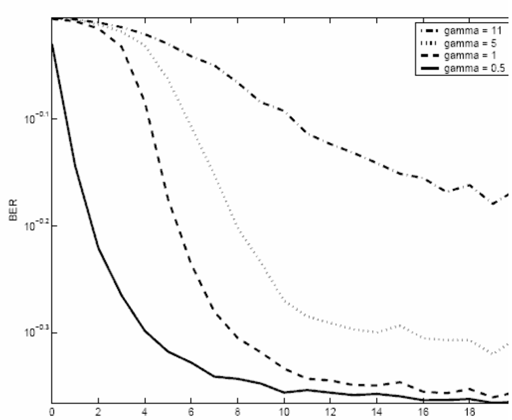

Fig. 4 BER versus SNR plots for LFLMS algorithm with dfferent γ values

Fig. 3 shows the MSE plot of LFLMS algorithm for different γ values for $\mu = 0.2$. From the figure we can conclude that MSE value increases with the increase in value of γ beyond μ. The error goes on increasing to an undesirable value if the γ value exceeds the maximum limit as given in (6). Fig. 4 shows the corresponding BER plots of LFLMS with the same γ values and same μ. The figure shows the results as expected from the increasing value of γ.

5. CONCLUSION

In this paper, a new algorithm, leaky FLMS (LFLMS), is proposed and the performance characteristics of the proposed algorithm in terms of MSE and BER plots are observed.. Although LFLMS algorithm has better convergence with respect to FLMS algorithm, the computational complexity increase slightly because the leaky factor forces the filter taps to keep learning in all of the frequency bands constantly and prevents the noise from building up. The convergence characteristic of LFLMS algorithm for different values of γ is also investigated. However, the convergence properties of the proposed LFLMS algorithm with correlated data are not shown here, which the authors are currently investigating. It is suggested to use this algorithm for frequency domain long memory applications like channel equalization, echo cancelation etc.

REFERENCES

[1] B. Widrow, J. M. McCool, M. G. Larimore, and C. R. Johnson, Jr., "Stationary and Nonstationary Learning Characteristics of the LMS Adaptive Filter", Proc. IEEE, vol. 64, no. 8, Aug. 1976.

[2] H. V. Poor and G. W. Wornell, Wireless Communications: Signal Processing Perspectives, Prentice-Hall: Upper Saddle River, NJ, 1998.

[3] R. R. Bit mead and B. D. O. Anderson, "Performance of Adaptive Estimation Algorithms in Dependent Random Environment," IEEE Trans. Automat. Contr., vol. 25, no. 4, pp. 788-794, 1980.

[4] M. Dentino, J. McCool, and B . Widrow, "Adaptive filtering in the Frequency Domain", Proc. IEEE, vol. 66, no. 12, pp. 1658-1659, Dec. 1978.

[5] G. A. Clark, S. K. Mitra, and S. R. Parker, "Block Implementation of Adaptive Digital Filters", IEEE Trans. Acoust., Speech, Signal Processing (Part III), vol. ASSP-29, pp. 747-752, June 1981.

[6] S. S. Narayan and A. M. Peterson, "Frequency Domain Least Mean Square Algorithm", Proc. IEEE, vol. 69, vol. 1, Jan. 1981.

[7] Simon Haykin, Adaptive Filter Theory, 4th Ed., Englewood Cliffs, Prentice Hall, 2002.

[8] N. Benvenuto and G. Cherubuni, "Algorithms for Communications Systems and Their Applications," Wiley, 2004.