A New Implementation of the LMS Algorithm as a Decision-directed Adaptive Equalizer with Decoding Delay

Sang-Sik Ahn*

ABSTRACT

This paper deals with the application of the LMS algorithm as a decision-directed adaptive equalizer in a communication receiver which also employs a sophisticated decoding scheme such as the Viterbi algorithm, in which the desired signal, hence the error, is not available until several symbol intervals later because of decoding delay. In such applications the implemented weight updating algorithm becomes DLMS and major penalty is reduced convergence speed. Therefore, every effort should be made to keep the delay as small as possible if it is not avoidable. In this paper we present a modified implementation in which the effects of the decoding delay can be avioded and perform some computer simulations to check the validity and the performance of the new implementation.

I. Introduction

The importance of adaptive channel equalization in high speed data transmission is well known [1]. The LMS type algorithms have been used in systems employing threshold detectors, in which no decision delay is involved and convergence analyses for this type of algorithms have been studied for a long time. For some time now, interest has been shown in using detectors that perform better than symbol-by-symbol detectors. For example, the Viterbi algorithm [2]-[4] provides a maximum likelihood sequence estimate and when the channel is known it performs better than conventional detectors at the expense of complexity. This complexity increases exponentially with the channel impulse response duration. Therefore, in order to reduce the complexity of the detector, an LMS type adaptive equalizer may be used to reduce the time dispersion of the channel impulse response [5]-[7]. However, the LMS algorithm can be implemented only under the assumption that we can measure the error signal and input vector at every iteration. Thus, when we employ a decision-directed adaptive equalizer if we use a decoding procedure such as Viterbi algorithm the desired signal, hance the error, is not available until several symbol intervals later because of decoding delay. In such applications the implemented weight updating algorithm becomes

$$W(n+1) = W(n) + \mu\{u(n-d) - X(n-d)^T W(n-d)\} X(n-d), \tag{1}$$

where W(n) is the weight vector for the n-th iteration cycle and μ is a step size which controls the rate of convergence, and u(n) is the desired signal. This is the modified version of the LMS algorithm known as the delayed LMS(DLMS) in the literature. Fig. 1 shows the direct implementation of the algorithm (1) employed as a decision directed adaptive equalizer with decoding delay d. It is well known that the major penalties of the delayed update are reduced convergence speed and, for non-stationary data, tracking capability [1]. Therefore, every effort should still be made to keep the delay as small as possible if it is not avoidable. In this paper, we concentrate on the application of the LMS algorithm as a decision-directed adaptive equalizer with decoding delay and propose a new way of implementation of an adaptive equalizer, in which the delayed weight updating can be avoided.

II. Derivation of the New Implementation of the Decision-Directed Adaptive Equalizer with Decoding Delay

To derive a new implementation of the decision-

*Dept. of Electrical and Information Engr., Korea Univ.
Manuscript Received: April 22, 1996.

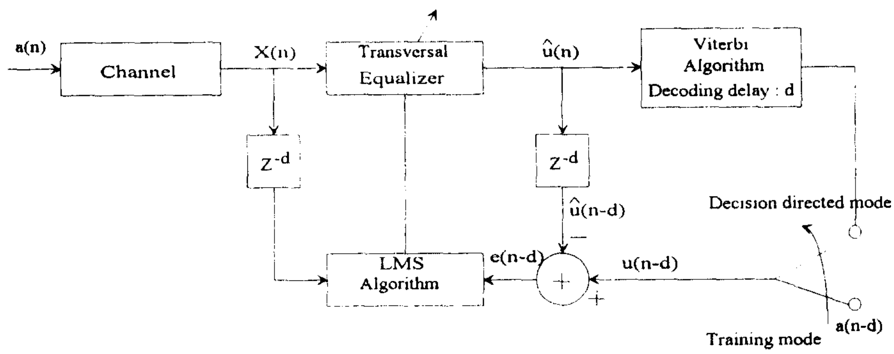

Fig 1. Direct implementation of a decision directed adaptive equalizer

directed adaptive equalizer with decoding delay, we first consider the DLMS algorithm in terms of the weight error vector $Y(n)$, which is defined as $Y(n) = W(n) - W_{opt}$, then we can rewrite (1) as

$$Y(n+1) = Y(n) - \mu X(n-d) X(n-d)^T Y(n-d) + \mu\{u(n-d) - X(n-d)^T W_{opt}\} X(n-d). \quad (2)$$

If we assume that perfect modeling is possible, then $u(n) = X(n)^T W_{opt}$ and (2) reduces to the following homogeneous recursive equation:

$$Y(n+1) = Y(n) - \mu X(n-d) X(n-d)^T Y(n-d). \quad (3)$$

Furthermore, to take full advantage of the measurable, if we use time varying step size $\mu(n)$ instead of constant μ, then (3) becomes

$$Y(n+1) = Y(n) - \mu(n) X(n-d) X(n-d)^T Y(n-d). \quad (4)$$

This is the homogeneous DLMS algorithm in terms of the weight error vector $Y(n)$. We will modify the algorithm (4) to the LMS by replacing $Y(n-d)$ with $Y(n)$ using the measurable as follows. First, we measure the quantity $X(n-d)^T Y(n-d)$ by measuring error $e(n-d)$ and we may add one more filter into the direct implementation to calculate $X(n-d)^T W(n)$. Then, at every iteration, we calculate

$$\frac{X(n-d)^T W(n) - u(n-d)}{e(n-d)}$$

and

$$\mu(n) = \mu \frac{X(n-d)^T W(n) - u(n-d)}{e(n-d)} \quad (5)$$

where μ is to be chosen to guarantee the stability and to control the rate of convergence. In the homogeneous case (5) can be written as

$$\mu(n) = \mu \frac{X(n-d)^T Y(n)}{X(n-d)^T Y(n-d)}. \quad (6)$$

Now, substituting (6) into (4) we have

$$Y(n+1) = Y(n) - \mu X(n-d) X(n-d)^T Y(n) \quad (7)$$

$$= \{I - \mu X(n-d) X(n-d)^T\} Y(n). \quad (8)$$

This is the standard LMS algorithm except the delay in the input. Finally, to obtain the implementation form, recalling $Y(n) = W(n) - W_{opt}$ we have

$$W(n+1) = W(n) - \mu X(n-d) X(n-d)^T \{W(n) - W_{opt}\} = W(n) - \mu X(n-d)\{X(n-d)^T W(n) - u(n-d)\}. \quad (9)$$

Fig. 2 is the modified implementation of the decision directed adaptive equalizer with decoding delay. As we can see from the computer simulation, in the modified implementation maximum allowable step size is not reduced by the inclusion of delay. Thus the modified implementation results in a faster convergence than the direct implementation, for a non-stationary channel, which implies better tracking capability.

Ⅲ. Computer Simulation of the Modified Algorithm

We study the use of LMS algorithm for adaptive equalization of a linear dispersive channel that produces unknown distortion and compare the performance of two realizations Fig. 1 and Fig. 2. Here

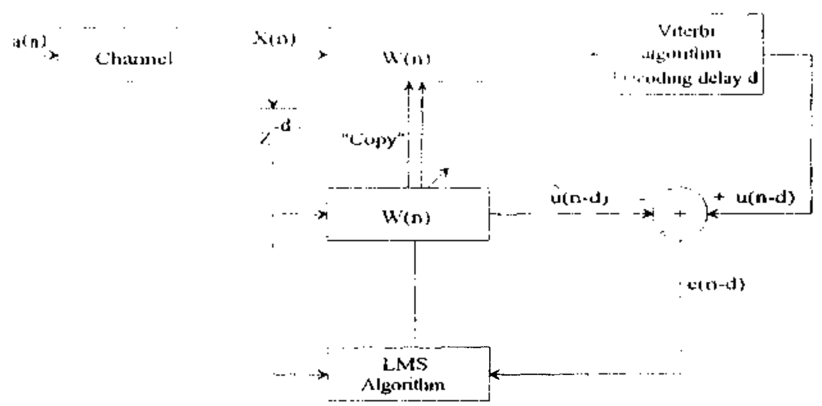

Fig 2. Modified implementation of a decision directed adaptive equalizer

we assume that the data are all real valued. Fig. 3 shows the block diagram of the system used to carry out the experiment, which is a modification of Haykin's model [8] that is widely accepted for simulation of the adaptive equalizer. The random data generator provides the test signal, a(n), used for probing the channel, whereas the random noise generator v(n) serves as the source of additive white noise that corrupts the channel output. These two random number generators are independent of each other. The adaptive equalizer has the task of correcting for the distortion produced by the channel in the presence of the additive white noise. The random signal generator, after suitable delay, also supplies the desired response applied to the adaptive equalizer.

The random sequence $\{a(n)\}$ applied to the channel input is in polar form, i.e., $a(n) = \pm 1$, so the sequence $\{a(n)\}$ has zero mean and unity variance. The impulse response of the channel is described by the raised cosine:

$$h(n) = \begin{cases} \frac{1}{2}\left[1 + \cos\left(\frac{2\pi}{\Theta}(n-2)\right)\right]; & n = 1, 2, 3 \\ 0; & \text{otherwise.} \end{cases} \tag{10}$$

where the parameter Θ controls the eigenvalue spread of the correlation matrix of the tap inputs in the equalizer, with the eigenvalue spread increasing with Θ. The sequence $\{v(n)\}$, produced by the second random generator, has zero mean and variance σ_v^2. The equalizer has $N = 11$ taps. Since the channel has an impulse response $\{h(n)\}$ that is symmetrical about time $n = 2$, it follows that the optimum tap weights $\{W_{opt}\}$ of the equalizer are likewise symmetric, about time $n = 5$. Therefore, the channel input $\{a(n)\}$ must be delayed by 7 symbols to provide the correct desired response for the equalizer. Finally, to simulate the decoding delay by *d* symbols we include Z^{-d}'s.

First, we investigate the effects of delay on the maximum allowable step size. In principle the Viterbi al-

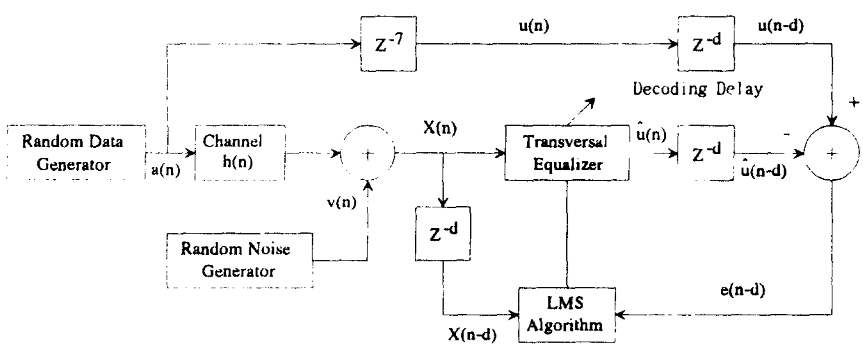

Fig 3. Configuration of the Simulation

gorithm can make a final decision on the initial state segment up to time n-τ when and only when all survivors at time n have the same initial state sequence segment up to time n-τ. The decoding delay τ is unbounded but is generally finite with probability one. In implementation, one actually makes a final decision after some fixed delay d, with d chosen large enough that the degradation due to premature decision is negligible, and it is typically of the order of 20 symbols or less [4]. Under the standard independence assumption, Kabal [9] showed that mean of W(n) in the DLMS algorithm converges to W_{opt} if and only if

$$0<\mu<\frac{2}{\lambda_{max}}\sin\frac{\pi}{2(2d+1)}, \tag{11}$$

where λ_{max} is the maximum eigenvalue of the input auto-correlation matrix of the equalizer. This condition reduces to that given by Widrow [10] for no delay. Thus, the delay in the weight updating procedure can be seen to reduce the maximum allowable step size by a factor $\sin\frac{\pi}{2(2d+1)}$. Furthermore, under the assumption that input is independent identically distributed with variance σ^2, we have another stability region for mean square convergence [11], [12]

$$0<\mu<\frac{2}{(N+2d)\sigma^2} \tag{12}$$

Again, the delay in the weight updating procedure can be seen to reduce the maximum allowable step size by a factor N/(N+2d). Note that maximum allowable step size depends on average input power as well as delay. Therefore, to investigate the effects of delay alone on the stability region, we need to remove the input dependence. To this end, we used a

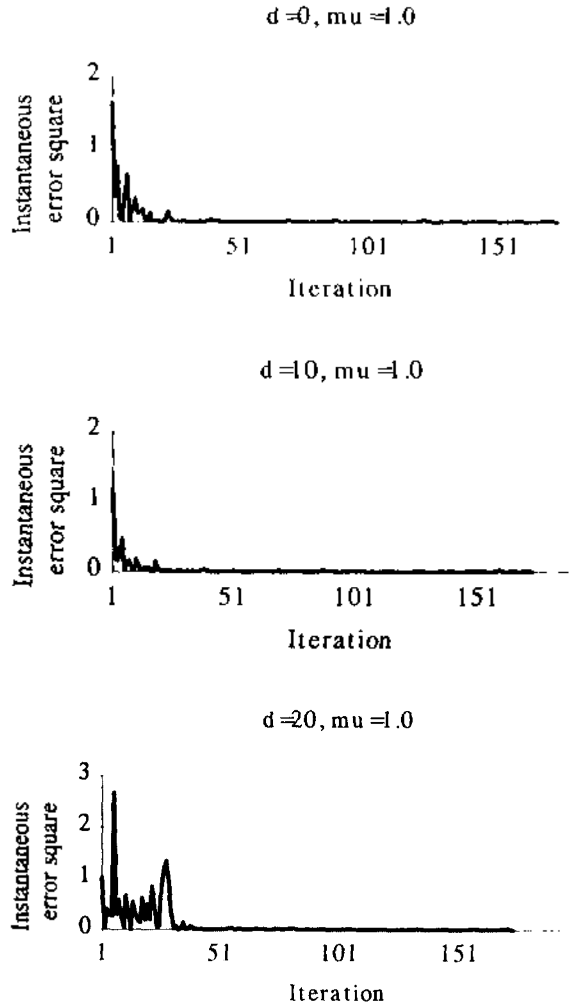

Fig 5. Independence of the performance of the modified DNLMS algorithm with varying delay

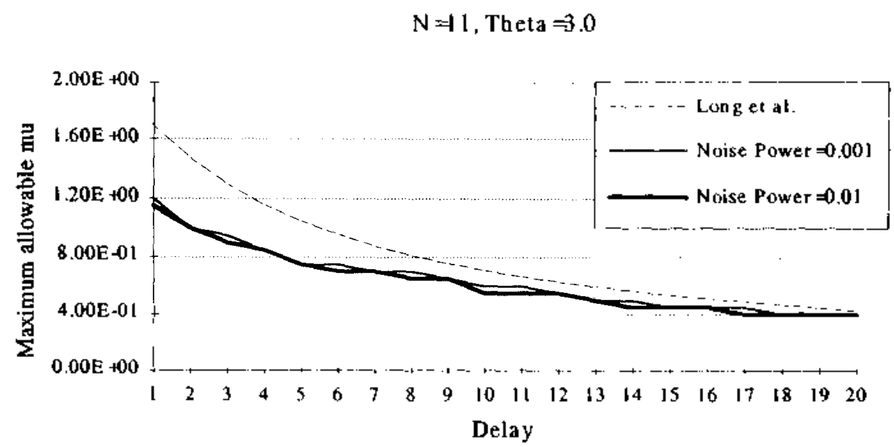

Fig 4. Maximum allowable step size of the DNLMS algorithm

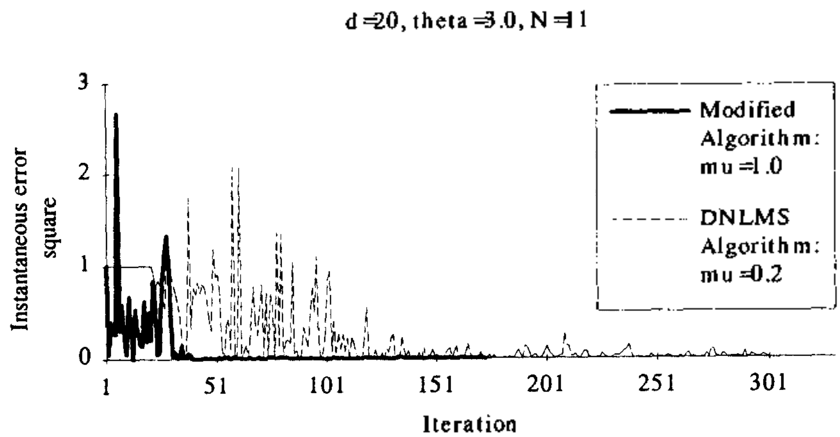

Fig 6. Convergence speed of the DNLMS and modified algorithm

normalized version of DLMS(DNLMS) as a weight updating algorithm in our computer simulation and in this case the stability bound (12) reduces to

$$0 < \mu < \frac{2N}{N+2d} . \quad (13)$$

In Fig. 4 we can see that the maximum allowable step size is decreasing with increasing delay and the above bound coincides reasonably well with the experiment. To test the convergence we adopt the following criterion: the algorithm converges if the ensemble-averaged error square, taken over 100 independent samples, is below 0.1 for 500 iterations. However, as shows in Fig. 5 the maximum allowable step size of the modified algorithm (9) is not changing at all even if the delay is introduced in the coefficient adaptation.

Finally, to compare the convergence speed of the two algorithms, we fix delay and the parameter Θ, and assign step size to $\mu = 0.2$ for the DNLMS algorithm and $\mu = 1.0$ for the modified algorithm such that they are located around the center of the allowable stability region. Fig. 6 shows the result of experiment in which we can observe that the modified algorithm converges faster than the DNLMS algorithm due to large available step size.

Ⅳ. Conclusion

In this paper we deal with the application of the LMS algorithm as a decision-directed adaptive equalizer in a communication receiver which also employs a sophisticated decoding scheme such as the Viterbi algorithm. In such systems, the LMS algorithm is constrained to operate with delay. We have derived modified implementation in which the effects of the feedback delay can be avoided, by including only one more set of multipliers, and presented some computer simulation results to check the validity and performance of the new implementation.

References

1. J. Proakis, "Digital Communications," McGraw-Hill, Inc., 1995.
2. G. D. Forney, Jr., "The Viterbi Algorithm," IEEE Proc. Vol. 61, No. 3, pp. 268-273, Mar. 1973.
3. G. D. Forney, Jr., "Maximum-likelihood sequence estimation of digital sequences in the presence of intersymbol interference," IEEE Trans. Inform. Theory, Vol. IT-18, No. 3, pp. 363-378, May 1972.
4. H. Kobayashi, "Correlative level coding and maximum likelihood decoding," IEEE Trans. Inform. Theory, Vol. IT-17, pp. 586-594, Sept. 1971.
5. S. U. H. Qureshi and E. E. Newhall, "An adaptive receiver for data transmission over time-dispersive channels," IEEE Trans. Inform. Theory, Vol. IT-19, pp. 448-457, July 1973.
6. D. D. Falconer and F. R. Magee, Jr., "Adaptive channel memory truncation for maximum likelihood sequence estimation," Bell Syst. Tech. J., Vol. 52, pp. 1541-1562, Nov. 1973.
7. A. Cantoni and K. Kwong, "Further results on the Viterbi algorithm equalizers," IEEE Trans. Inform. Theory, Vol. IT-20, pp. 764-767, Nov. 1974.
8. S. Haykin, "Adaptive Filter Theory," Prentice Hall, Englewood Cliffs, NJ, 1991.
9. P. Kabal, "The Stability of Adaptive Minimum Mean Square Error Equalizers Using Delayed Adjustment,"

IEEE Trans. on Comm., Vol. 31, No. 3, pp. 430-432 Mar. 1983.

10. B. Widrow and S. Stearns, "Adaptive Signal Processing," Prentice Hall, Englewood Cliffs, NJ, 1985.
11. G. Long, F. Ling, and J. Proakis, "The LMS Algorithm with Delayed Coefficient Adaptation," IEEE Trans. on ASSP, Vol. 37, No. 9, pp. 1397-1405, Sept. 1989.
12. G. Long, F. Ling, and J. Proakis, "Corrections to 'The LMS Algorithm with Delayed Coefficient Adaptation'," IEEE Trans. Signal Processing, Vol. 40, pp. 230-232, Jan. 1992.

▲Sang-Sik Ahn

Sang-Sik Ahn received the B.S. degree in electronic engineering and M.S. degree in electrical engineering from Korea University, Seoul in 1983 and 1985, respectively, and Ph.D. degree in electrical engineering from Polytechnic University in 1994.

Since 1995 he has been an Assistant Professor of Electrical and Information Engineering at Korea University, Seochang, Jochiwon. Prior to this, he had been with the GoldStar Central Research Lab. as a Senior Research Engineer from 1984 to 1987 and as a Principal Research Engineer from 1994 to 1995. From 1991 to 1993 he was a Teaching Fellow at Polytechnic University. His current research interests include adaptive signal processing, communication system, and VLSI implementation of DSP algorithms.