

The Asymptotic Analysis of the Smoothed Least Mean Square Algorithm and Its Applications

SLMS 알고리즘의 근사적 분석과 그 응용

Ikjoo Chung*

정 익 주*

Abstract

In this paper, we investigate the asymptotic performance of the smoothed least mean square(SLMS) algorithm, which gives another insight into the SLMS algorithm. Based on the results obtained, we consider the relation between the SLMS algorithm and other LMS-type algorithms-the linearly filtered gradient(LFG) algorithm, and the stochastic gradient descent (SGD) algorithm. In addition, it is shown that nonstationary performance of the SLMS algorithm is comparable with that of the LMS algorithm by computer simulation

요 약

Berman과 Feuer의 SLMS(smoothed least mean square) 알고리즘의 근사적 분석을 행하여 보다 유용한 분석결과를 얻었다. 수렴범위와 misadjustment에 대한 분석에서는 기존의 알고리즘의 분석결과들과 비교할 수 있는 형태로 얻었을뿐만 아니라 여러 변수들이 이 알고리즘의 성능에 미치는 영향을 명확히 알 수 있는 형태로 얻었다. 둘째로 몇몇 서로 유사한 알고리즘들을 비교검토함으로써 서로간의 관계를 밝히고 이 결과들을 해석하였다.

이어서 위의 분석결과들이 유효함을 실험을 통하여 밝혔다. 수렴한계 근처에서 LMS알고리즘보다 안정됨을 보였다. 이들 알고리즘의 비정상특성(nonstationary characteristics)에 대하여서도 살펴보았는데, SLMS알고리즘의 경우 추적능력의 별다른 희생 없이도 가중계수(weight)의 잡음을 줄일 수 있음을 보였다.

1. Introduction

Since Widrow and Hoff proposed the least mean square(LMS) algorithm[1][2]. Many researchers have studied various structures and adaptation algorithms for adaptive digital filters (ADF's). These ADF's have found many applications in situations where the statistics of input processes are unknown or changing. The ap-

plication areas of ADF's include line enhancing [4], noise cancelling[5], channel equalization[6], and adaptive array processing[7], and are being widened further with the rapid advance in digital IC technology. Among various structures and adaptation algorithms proposed so far, the finite impulse response(FIR) ADF using the LMS algorithm is being widely used due to its simplicity in realization. However, one drawback of the LMS algorithm is known to be its slow convergence speed when the input signal is highly correlated [2][3]. The convergence speed of the LMS algo-

*Department Electronics
Kangwon National University

접수일자: 1992. 12. 12.

ithm depends on the choice of the design factor μ called a convergence factor. To improve the convergence speed of the adaptive algorithm, many other filter algorithms have been proposed and investigated. One approach to increase the convergence speed is to use a matrix convergence factor. This type of weight adjustment is known as the self-orthogonalization[8]. Another approach is to use time varying and/or individual convergence factors. A modified LMS called α -LMS[9], where a convergence factor μ was made time varying in inverse proportion to the input power, was introduced.

The performance characteristics of the LMS algorithm have been studied extensively and are relatively well understood. Under the assumption of stationary uncorrelated input data vectors, the time constant of the LMS learning curve was shown to be inversely proportional to a convergence factor μ and the misadjustment to be directly proportional to it[2][3], and recently under the assumption of uncorrelated Gaussian input, exact analysis of the LMS algorithm including the second moment behavior was studied in [10].

The most of the aforementioned ADF algorithms based on the LMS algorithm were mainly intended to speed up the adaptive procedure with the guarantee of stability. Hence the performance analyses of these ADF algorithms show that the convergence speed can be improved but that those algorithms do not provide satisfactory misadjustment performance in comparison with that of the conventional LMS algorithm. Attempts to decrease misadjustment have also been made to devise more efficient algorithms by using two-stage method. Large values of μ are first chosen to achieve rapid convergence; then, after convergence has been achieved, smaller values are chosen to minimize misadjustment. Unfortunately, this method does not work if input is not stationary. In a recent paper[11], an adaptive damped convergence factor was suggested. There the new convergence factor was decreased as the gradient of perform-

ance surface approached to zero. Hence, the misadjustment was reduced when the algorithm was in the steady state. However, there is a limitation in reducing misadjustment using the decreasing convergence factor, for the misadjustment is essentially due to gradient noise[2][3]. In order to reduce the gradient noise, ensemble average should be taken and in an ergodic case time average can replace it. In the stochastic gradient descent (SGD) algorithm[12], the stochastic gradient of the time-averaged squared stochastic error was used for the estimation of the gradient of MSE. Using a long term-averaged gradient estimate gives a small misadjustment. But, the convergence speed to guarantee the stability gets decreased and the performance in a nonstationary case becomes poor.

Recently, the smoothed least means square (SLMS) algorithm was proposed and its performance was analyzed by Feuer and Berman[13]. Their study indicated that the steady state performance could be improved by adding a gradient smoothing element, while not affecting the convergence speed. In effect, the SLMS algorithm is a sort of the linearly filtered gradients (LFG) algorithm proposed previously by Proakis [14]. However, our analysis shows that the convergence behavior is quite different from that of the LFG algorithm in some aspects. On the other hand, the SLMS may be considered a modification of the SGD algorithm. In case of the SGD algorithm, a small misadjustment can be achieved at the expense of convergence speed. However, it can be shown that the SLMS algorithm gives a small misadjustment without the loss of convergence speed.

In this paper, we investigate the asymptotic performance of the SLMS algorithm, which gives another insight into the SLMS algorithm. Based on the results obtained, we consider the relation between the SLMS algorithm and other LMS-type algorithms-the linearly filtered gradient(LFG) algorithm, and the stochastic gradient

descent(SGD) algorithm. And we show that nonstationary performance of the SLMS is as good as that of the LMS algorithm by computer simulation.

The organization of this paper is as follows. In Section II, we formulate the SLMS algorithm and investigate its convergence and asymptotic performance. In section III, we consider the relationships between the SLMS algorithm and other algorithms. In Section IV, we present various computer simulations to verify the results obtained in Section II and III. Finally, we draw conclusions in Section V.

II. The Asymptotic Analysis of the SLMS Algorithm

The SLMS algorithm is given as [13][15]

$$\mathbf{w}_{n+1} = \mathbf{w}_n + \mu \Gamma_n \quad (1a)$$

$$\Gamma_n = \alpha \Gamma_{n-1} + (1-\alpha) e_n \mathbf{x}_n, \quad 0 < \alpha < 1 \quad (1b)$$

where \mathbf{w}_n and \mathbf{x}_n are the $(N \times 1)$ weight vector and the $(N \times 1)$ input vector, respectively. μ is a convergence factor, α is a smoothing factor, and the error signal e_n is defined as $e_n = d_n - y_n$, where d_n is a desired response and $y_n = \mathbf{x}_n^T \mathbf{w}_n$. It is well known that the optimal weight vector \mathbf{w}_{opt} is

$$\mathbf{w}_{opt} = \mathbf{R}_x^{-1} \mathbf{p}_x \quad (2)$$

where $\mathbf{R}_x = E[\mathbf{x}_n \mathbf{x}_n^T]$ and $\mathbf{p}_x = E[d_n \mathbf{x}_n]$.

Since the input autocorrelation matrix \mathbf{R}_x is symmetric, there exists a matrix \mathbf{Q}_x such that $\mathbf{Q}_x^T = \mathbf{Q}_x^{-1}$ and

$$\mathbf{Q}_x^{-1} \mathbf{R}_x \mathbf{Q}_x = \Lambda_x = \text{diag}(\lambda_i) \quad (3)$$

where t denotes a transpose operator and $\lambda_i, i = 1, 2, \dots, N$ are the eigenvalues of \mathbf{R}_x . It can be easily shown that (1b) is rewritten as

$$\Gamma_n = (1-\alpha) \sum_{i=0}^{\infty} \alpha^i e_{n-i} \mathbf{x}_{n-i} \quad (4)$$

From (4), the gradient estimate used in the SLMS algorithm includes an infinite summation of exponentially weighted data with the present data contributing more ingeneral, while in the SGD algorithm equally weighted finite data are used for a gradient estimate[12]. In addition, in the SLMS algorithm, a weight vector is adjusted at every instant unlikely the SGD algorithm. From these facts, we may find the possibility that this algorithm has almost the same convergence speed as that of the LMS algorithm and at the same time gives smaller misadjustment. We will show later that the above statement is justified through analysis and computer simulations.

For the proof that the mean weight vector converges to the optimum Wiener solution, We also assume that the input sequences are statistically independent over time and stationary, and have zero mean. Using this assumption, we can obtain a difference equation for the mean weight vector by averaging (1a) :

$$\begin{aligned} E[\mathbf{w}_{n+1}] &= E[\mathbf{w}_n] + \mu(1-\alpha) \sum_{i=0}^{\infty} \alpha^i E[d_{n-i} \mathbf{x}_{n-i}] - E[\mathbf{x}_{n-i} \mathbf{x}_{n-i}^T E[\mathbf{w}_{n-i}]] \\ &= E[\mathbf{w}_n] + \mu(1-\alpha) \sum_{i=0}^{\infty} \alpha^i \{ [\mathbf{p}_x - \mathbf{R}_x E[\mathbf{w}_{n-i}]] \} \\ &= E[\mathbf{w}_n] + \mu \mathbf{p}_x (1-\alpha) \sum_{i=0}^{\infty} \alpha^i - \mu(1-\alpha) \mathbf{R}_x \sum_{i=0}^{\infty} \alpha^i E[\mathbf{w}_{n-i}] \\ &= E[\mathbf{w}_{n+1}] \end{aligned} \quad (5)$$

Using $\mathbf{w}_{opt} = \mathbf{R}_x^{-1} \mathbf{p}_x$ and $(1-\alpha) \sum_{i=0}^{\infty} \alpha^i = 1$, (5) becomes

$$E[\mathbf{w}_{n+1}] = E[\mathbf{w}_n] + \mu \mathbf{R}_x \mathbf{w}_{opt} - \mu(1-\alpha) \mathbf{R}_x \sum_{i=0}^{\infty} \alpha^i E[\mathbf{w}_{n-i}] \quad (6)$$

Defining $\mathbf{v}_n = \mathbf{w}_n - \mathbf{w}_{opt}$, we have

$$\begin{aligned} E[\mathbf{v}_{n+1}] &= E[\mathbf{v}_n] - \mu \mathbf{R}_x \{ (1-\alpha) \sum_{i=0}^{\infty} \alpha^i E[\mathbf{w}_{n-i}] - \mathbf{w}_{opt} \} \\ &= E[\mathbf{v}_n] - \mu \mathbf{R}_x \{ (1-\alpha) \sum_{i=0}^{\infty} \alpha^i E[\mathbf{w}_{n-i}] - (1-\alpha) \sum_{i=0}^{\infty} \alpha^i \mathbf{w}_{opt} \} \\ &= E[\mathbf{v}_n] - \mu \mathbf{R}_x (1-\alpha) \sum_{i=0}^{\infty} \alpha^i \{ E[\mathbf{w}_{n-i}] - \mathbf{w}_{opt} \} \end{aligned}$$

$$= E[\mathbf{v}_n] - \mu \mathbf{R}_x (1 - \alpha) \sum_{i=0}^{\infty} \alpha^i E[\mathbf{v}_{n-i}] \quad (7)$$

Let (7) be premultiplied by \mathbf{Q}_x^{-1}

$$E[\mathbf{Q}_x^{-1} \mathbf{v}_{n+i}] = E[\mathbf{Q}_x^{-1} \mathbf{v}_n] - \mu \mathbf{Q}_x^{-1} \mathbf{R}_x \mathbf{Q}_x (1 - \alpha) \sum_{i=0}^{\infty} \alpha^i E[\mathbf{Q}_x^{-1} \mathbf{v}_{n-i}] \quad (8)$$

Defining $\tilde{\mathbf{v}}_n = \mathbf{Q}_x^{-1} \mathbf{v}_n$ again, (8) may be rewritten as

$$E[\tilde{\mathbf{v}}_{n+1}] = E[\tilde{\mathbf{v}}_n] - \mu \Lambda_x (1 - \alpha) \sum_{i=0}^{\infty} \alpha^i E[\tilde{\mathbf{v}}_{n-i}] \quad (9a)$$

$$= \{I - \mu(1 - \alpha) \Lambda_x\} E[\tilde{\mathbf{v}}_n] - \mu \Lambda_x (1 - \alpha) \sum_{i=1}^{\infty} \alpha^i E[\tilde{\mathbf{v}}_{n-i}] \quad (9b)$$

$$= \{I - \mu(1 - \alpha) \Lambda_x\} E[\tilde{\mathbf{v}}_n] - \mu \Lambda_x (1 - \alpha) \alpha \sum_{i=0}^{\infty} \alpha^i E[\tilde{\mathbf{v}}_{n-i-1}] \quad (9c)$$

In (9a), replacing index n with $n-1$, we have

$$E[\tilde{\mathbf{v}}_n] = E[\tilde{\mathbf{v}}_{n-1}] - \mu \Lambda_x (1 - \alpha) \sum_{i=0}^{\infty} \alpha^i E[\tilde{\mathbf{v}}_{n-i-1}] \quad (10)$$

Manipulating (10), we obtain the following equation:

$$- \mu \Lambda_x (1 - \alpha) \alpha \sum_{i=0}^{\infty} \alpha^i E[\tilde{\mathbf{v}}_{n-i-1}] = \alpha \{E[\tilde{\mathbf{v}}_n] - E[\tilde{\mathbf{v}}_{n-1}]\} \quad (11)$$

From (9c) and (11), it can be shown that it can be shown that $E[\tilde{\mathbf{v}}_n]$ satisfies the following the second order recursive equation:

$$E[\tilde{\mathbf{v}}_{n+1}] = \{(1 + \alpha)\} I - \mu(1 - \alpha) \Lambda_x \} E[\tilde{\mathbf{v}}_n] - \alpha E[\tilde{\mathbf{v}}_{n-1}] \quad (12)$$

For changing (12) into a first order equation of dimension $2N$, let

$$\mathbf{A}_n = \begin{bmatrix} E[\tilde{\mathbf{v}}_n] \\ E[\tilde{\mathbf{v}}_{n+1}] \end{bmatrix} \quad (13a)$$

$$\mathbf{X} = -\alpha I \text{ and } \mathbf{Y} = (1 + \alpha)I - \mu(1 - \alpha) \Lambda_x. \quad (13b)$$

Then, (12) may be expressed as a first-order

equation:

$$\mathbf{A}_n = \mathbf{C} \mathbf{A}_{n-1} \quad (14)$$

where

$$\mathbf{C} = \begin{bmatrix} \mathbf{O} & \mathbf{I} \\ \mathbf{X} & \mathbf{Y} \end{bmatrix} \quad (15)$$

In (15), \mathbf{O} denotes $(N \times N)$ zero matrix. The necessary and sufficient condition for convergence of the vector \mathbf{A}_n is that the eigenvalues of \mathbf{C} be less than unity in absolute value. Now we consider the convergence condition in connection with the convergence factor μ and the smoothing factor α . Let η be any eigenvalue of \mathbf{C} with the corresponding eigenvector $\mathbf{q}' = [\mathbf{q}_1', \mathbf{q}_2']$ where \mathbf{q}_1 and \mathbf{q}_2 are N -dimensional column vectors. Applying these to \mathbf{C} of (14),

$$\mathbf{C} \mathbf{q}' = \eta \mathbf{q}' \quad (16)$$

which reduces to

$$\mathbf{X} \mathbf{q}_1 + \eta \mathbf{Y} \mathbf{q}_1 = \eta^2 \mathbf{q}_1 \quad (17)$$

or, equivalently

$$-\alpha \mathbf{q}_1 + \eta \{(1 + \alpha)I - \mu(1 - \alpha) \Lambda_x\} \mathbf{q}_1 = \eta^2 \mathbf{q}_1. \quad (18)$$

Again, (18) can be expressed in a scalar equation:

$$\eta^2 - \{(1 + \alpha) - \mu(1 - \alpha) \lambda_i\} \eta + \alpha = 0 \quad \text{for all } i, \quad (19)$$

For convergence, the absolute value of η should be less than unity. For a particular λ_i , it is shown that a necessary and sufficient condition for $|\eta| < 1$ in (19) is

$$0 < \eta < \frac{1 + \alpha}{1 - \alpha \lambda_i} \quad (20)$$

Since the inequality should be satisfied for all λ_i , the convergence condition becomes

$$0 < \eta < \frac{1 + \alpha}{1 - \alpha \lambda_{\max}} \quad (21)$$

That is, if μ satisfies the above convergence condition, $E[\tilde{v}_n]$ converges to zero and there by $E[\tilde{v}_n]$ also converges to zero. Subsequently, $E[\tilde{w}_n]$ converges to w_{opt} since $\tilde{v}_n = \tilde{w}_n - w_{opt}$. Moreover, the above inequality is the same form as the convergence conditions for other LMS type algorithms and is simpler and more useful for understanding and comparison with others than that presented in [13].

Next, we will derive the misadjustment of the SLMS algorithm asymptotically by following the same strategy of derivation of the LMS algorithm presented in [2][3]. From (1a) and (4) the smoothed gradient estimate $\nabla \hat{e}_n$ is

$$\nabla \hat{e}_n = -2(1-\alpha) \sum_{i=0}^L \alpha^i e_{n-i} x_{n-i} \quad (22)$$

Since gradient noise in steady-state is equal to gradient estimate [2][3], that is, $\tilde{N}_n \approx \nabla \hat{e}_n$, the covariance of gradient noise is

$$\text{cov}[\tilde{N}_n] = E[\tilde{N}_n \tilde{N}_n^T] \approx 4(1-\alpha)^2 \sum_{i=0}^L \sum_{j=0}^L \alpha^i \alpha^j E[e_{n-i} e_{n-j} x_{n-i} x_{n-j}^T] \quad (23)$$

where \tilde{N}_n denotes the gradient noise in steady-state. For analysis, we assume as earlier researchers did [12][16] that the sequence of vectors $\{x_n, d_n^T\}$ is a zero-mean jointly Gaussian process and uncorrelated over time, and then

$$E[e_{n-i} x_{n-j}] = 0, \text{ for } i \neq j \quad (24)$$

Furthermore, after the system is adapted sufficiently long, we can assume the system is in steady-state at any instants $n-i$ and $n-j$, even for considerably large values of i and j , and therefore the orthogonal principle can be applied, i.e.,

$$E[e_{n-i} x_{n-i}] = 0 \text{ and } E[e_{n-i}^2] = \epsilon_{\min}, \text{ for } i=0,1,2,\dots,L \quad (25)$$

where L is a considerably large value. Moreover,

terms for $i > L$ and $j > L$ in (23) are approximately zero since α is small in practical circumstances. From (24) and (25), since orthogonality principle implies uncorrelatedness for a zero-mean jointly gaussian process, e_{n-i} and x_{n-j} are uncorrelated for $i, j = 0, 1, 2, \dots, L$ and the terms for $i > L$ and $j > L$ can be ignored. Consequently, (23) may be rewritten as

$$\begin{aligned} \text{cov}[\tilde{N}_n] &\approx 4(1-\alpha)^2 \sum_{i=0}^L \sum_{j=0}^L \alpha^i \alpha^j E[e_{n-i} e_{n-j}] E[x_{n-i} x_{n-j}^T] \\ &= 4(1-\alpha)^2 R_x \sum_{i=0}^L \alpha^{2i} E[e_{n-i}^2] \\ &= 4(1-\alpha)^2 R_x \epsilon_{\min} \sum_{i=0}^L \alpha^{2i} \\ &= 4 \frac{(1-\alpha)}{(1+\alpha)} R_x \epsilon_{\min} \end{aligned} \quad (26)$$

Then the covariance of the projection of the gradient noise, $\tilde{N}_n = Q_x^{-1} \tilde{N}_n$, becomes

$$\text{cov}[\tilde{N}_n] \approx 4 \frac{1+\alpha}{1-\alpha} \Lambda_x \epsilon_{\min} \quad (27)$$

Referring to [2] and [3], we can obtain from (27)

$$\text{cov}[\tilde{v}_n] \approx \frac{1}{2} \frac{1+\alpha}{1-\alpha} \mu \epsilon_{\min} I \quad (28)$$

and the averaged excess mean squared error $E[\epsilon_s]$ is given as

$$E[\epsilon_s] \approx \frac{1}{2} \frac{1+\alpha}{1-\alpha} \mu \epsilon_{\min} \text{tr} R_x \quad (29)$$

where tr denotes trace operator. Accordingly, the misadjustment M is

$$M = \frac{E[\epsilon_s]}{\epsilon_{\min}} \approx \frac{1}{2} \frac{1+\alpha}{1-\alpha} \mu \text{tr} R_x \quad (30)$$

Comparing (30) with the misadjustment of the LMS algorithm presented in [2], one can see that the misadjustment of the SLMS algorithm is smaller than that of the LMS algorithm since α is greater than zero. Moreover, when $\alpha=0$, (30) reduces to the misadjustment of the LMS algorithm. Once the system is adapted long enough to make it possible to set a time index n to be infinity, it is plausible

that the system is still in steady-state at instants $n-i$ and $n-j$, even if the summation indices i and j are arbitrarily large. Therefore, though $\alpha \approx 1$, (30) is valid, and the misadjustment can become arbitrarily small as the time index n goes to infinity if input data are stationary

Now, we consider the convergence speed of the SLMS algorithm. For the LMS algorithm, it can be seen that the convergence speed of the LMS algorithm depends on the diagonal matrix $(I - \mu \Lambda_r)$. On the other hand, based on (14), the convergence speed of the SLMS algorithm depends on the eigenvalue of the matrix C . For the comparison of the convergence speeds between the SLMS and LMS algorithms, we consider a particular eigenvalue λ_i of R_x . In (19), for the particular λ_i it can be shown that if the equation (19) has real roots (In practical applications, this assumption is valid. See Appendix.), the absolute value of the corresponding η is

$$|\eta| := \left| \frac{(1+\alpha) - \mu(1-\alpha)\lambda_i + \sqrt{(1-\alpha)^2 - 2\mu\lambda_i(1-\alpha)(1+\alpha) + \mu^2\lambda_i^2(1-\alpha)^2}}{2} \right| \quad (31)$$

In (31), we chose the larger of the two real roots since between the corresponding two real roots η_1 and η_2 , the larger one governs the convergence speed. Under the assumption $\frac{1}{2}\mu\lambda_i \ll 1$ for all i , which was also assumed in [2] and [3], since $\mu^2\lambda_i^2$ is very small and $0 < \alpha < 1$, the equation inside the square root in (31) can be approximated as follows:

$$\begin{aligned} & (1-\alpha)^2 - 2\mu\lambda_i(1-\alpha)(1+\alpha) + \mu^2\lambda_i^2(1-\alpha)^2 \\ & \approx (1-\alpha)^2 - 2\mu\lambda_i(1-\alpha)(1+\alpha) + \mu^2\lambda_i^2(1+\alpha)^2 \\ & = \{(1-\alpha) - \mu\lambda_i(1+\alpha)\}^2 \end{aligned} \quad (32)$$

Inserting (32) into (31), we have

$$|\eta| \approx |1 - \mu\lambda_i| \quad (33)$$

From (33), one can see that the convergence

speed of the SLMS algorithm is approximately equal to that of the LMS algorithm in practical applications. This result is consistent with that obtained by [13], that is, the steady-state performance is improved while the convergence speed is the same as that of the LMS algorithm.

The equation (19) may have complex roots. In practical applications, however, this is a rare case which results in a damped oscillation. As mentioned in appendix, the assumption,

$$\frac{1}{2}\mu\Lambda_x \ll I, \text{ guarantees that (19) has real roots.}$$

III. The consideration on the Relation with Other LMS-type Algorithms

Here, the comparison between the SLMS algorithm and other algorithms-the SGD algorithm and the LFG algorithm-will be made. First, in the SGD algorithm the stochastic gradient of the time-averaged squared error is used as follows [12]:

$$w_{(n+1)} = w_{nk} - \frac{1}{2}\mu \nabla \hat{e}_{nk} \quad (34a)$$

$$\nabla \hat{e}_{nk} = -\frac{2}{K} \sum_{q=0}^{K-1} e_{nk+q} x_{nk+q} \quad (34b)$$

$$w_{nk+q} = w_{nk}, \quad q = 1, 2, \dots, K-1 \quad (34c)$$

For the special case $K=1$, the SGD algorithm is the very LMS algorithm. Furthermore, $K=1$ yields the most rapidly converging algorithm. The SLMS algorithm is different from the SGD algorithm in that the former has a gradient estimate averaged with an exponential window instead of a rectangular window and adjusted at every instant. However, one can expect that steady-state behaviors of these two algorithms will be similar. As for the SGD algorithm, it is known [12] that the misadjustment is

$$M \approx \frac{1}{2K} \mu \text{tr} R_x \quad (35)$$

That is, averaging gradient estimate over many

time instants gives small misadjustment. Under the assumption with respect to the desired response and input vector, $e_n \mathbf{x}_n$ used in (22) is independent process in steady-state, and then it can be shown that [17]

$$-2(1-\alpha) \sum_{i=0}^{\infty} \alpha^i e_{n-1} \mathbf{x}_{n-1} = -\frac{2}{LP} \sum_{i=0}^{LP} e_{n-i} \mathbf{x}_{n-i} \quad (36a)$$

where LP is called a learning period and given as

$$LP = \frac{1+\alpha}{1-\alpha} \quad (36b)$$

One can see that the right side in (36a) is the very gradient estimate used in the SGD algorithm (See(34b)), and furthermore, replacing \mathbf{K} in (35) with (36b) results in (30). This fact is in agreement with our analysis as we expected.

In the next place, the LFG algorithm proposed by [14] uses a linearly filtered gradient

$$\nabla \hat{e}_n = \alpha \nabla \hat{e}_{n-1} + (1-\alpha) e_n \mathbf{x}_n, \quad 0 < \alpha < 1 \quad (37)$$

The N -dimensional filter is in effect a set of N identical first-order filters operating in parallel. Each has a z -transform

$$H_f(z) = \frac{1}{1-\alpha z^{-1}} \quad (38)$$

In case of the SLMS algorithm, a different filter which has a z -transform expressed below is used :

$$H_s(z) = \frac{1-\alpha}{1-\alpha z^{-1}} \quad (39)$$

According to [14], the condition that μ and α must satisfy for stability is

$$0 < \mu < (1+\alpha) \frac{2}{\lambda_{\max}} \quad (40)$$

The above condition is consistent with our result in that the LFG algorithm also has the advantage of extending the upper bound on the value of μ as in the case of the SLMS algorithm. However, as

for the misadjustment, these two algorithms show very different convergence characteristics. According to [14], the misadjustment of the LFG algorithm is

$$M \approx \frac{1}{2(1-\alpha)} \mu \text{tr} \mathbf{R}_x \quad (41)$$

Comparing (41) with (30), increasing the constant α means a large misadjustment in contrast to the SLMS algorithm. These conflicting results may be explained as follows : The LFG algorithm may be rewritten as

$$\mathbf{w}_{n+1} = \mathbf{w}_n + \mu e_n \mathbf{x}_n + \alpha(\mathbf{w}_n - \mathbf{w}_{n-1}) \quad (42)$$

This expression is the very algorithm called the momentum LMS (MLMS), which was proposed recently by [18]. Referring to the analysis in [18], in which the same approach as presented in [16] was followed, it can also be shown that increasing the constant α which was called momentum constant there makes the misadjustment large. The SLMS algorithm may also be expressed in the form of the equation (42) as follows

$$\mathbf{w}_{n+1} = \mathbf{w}_n + \hat{\mu} e_n \mathbf{x}_n + \alpha(\mathbf{w}_n - \mathbf{w}_{n-1}) \quad (43a)$$

where

$$\hat{\mu} = \mu(1-\alpha) \quad (43b)$$

From (43b), the larger the constant α , the smaller the virtual convergence factor $\hat{\mu}$, and therefore we can interpret the conflicting fact mentioned above as follows : the effect of the virtual convergence factor $\hat{\mu}$ is more dominant than that of momentum term $\alpha(\mathbf{w}_n - \mathbf{w}_{n-1})$, as the constant α becomes large, for a small convergence factor gives a small misadjustment.

Now, we consider the convergence speed of the LFG algorithm with the same strategy as presented in Section II. In case of the LFG algorithm, the analogous equation with (19) is

$$\eta^2 - \{(1+\alpha) - \mu\lambda_i\}\eta + \alpha = 0, i = 0, 1, 2, \dots, N-1 \quad (44)$$

The absolute value of the larger one of two roots in (44) is given as

$$|\eta| = \left| \frac{\{(1+\alpha) - \mu\lambda_i\} + \sqrt{(1+\alpha)^2 - 2(1+\alpha)\mu\lambda_i + \mu^2\lambda_i^2 - 4\alpha}}{2} \right| \quad (45)$$

The equation inside the square root in (45) can be rewritten as follows :

$$\begin{aligned} (1+\alpha)^2 - 2(1+\alpha)\mu\lambda_i + \mu^2\lambda_i^2 - 4\alpha \\ = (1-\alpha)^2 - 2(1-\alpha)\mu\lambda_i + \mu^2\lambda_i^2 - 4\alpha\mu\lambda_i \\ = \{(1-\alpha) - \mu\lambda_i\}^2 - 4\alpha\mu\lambda_i \end{aligned} \quad (46)$$

From (45) and (46), we have

$$|\eta| < |1 - \mu\lambda_i| \quad (47)$$

The equation (47) indicates that the convergence speed of the LFG algorithm is faster than that of the LMS algorithm. In addition, since large value of α means small value of $|\eta|$, the larger the value of α is, the faster the convergence speed is.

IV. Computer Simulation Results and Discussion

In the previous sections, we have studied the asymptotic convergence behavior of the SLMS algorithm and compared the results with those of the results with those of other LMS-type algorithms. In this section, by computer simulation, we verify the analytical results and investigate nonstationary characteristics. For our computer simulation, we consider a channel equalizer and an adaptive line enhancer (ALE) problems.

In the first place, a channel equalizer is considered. The receiver of a simplified base-band binary data transmission system is shown in Fig. 1. This adaptive equalizer and channel characteristics are identical to those used in [19]. The channel im-

pulse response used in our simulation is given by in (48) and W has been set to 3.3 for eigenvalue ratio of 21.

$$h_i = \begin{cases} 1/2[1 - \cos\{2\pi(i-2/W)\}], & 1 \leq i \leq 3 \\ 0, & \text{otherwise} \end{cases} \quad (48)$$

The output of the channel has been scaled to unity power. White Gaussian noise with variance of 0.001 has been added to the equalizer input. The number of the equalizer weights was 16 and all the initial values of them were set to zero. As a performance measure of the ADF, we have used the MSE estimate that is obtained by computing an ensemble average of 500 individual squared error. First, we discuss the results of computer simulation on the convergence speed and

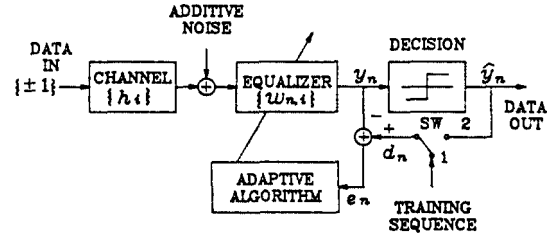


Fig. 1. Base-band data transmission system using an adaptive channel equalizer

misadjustment. Fig. 2. shows the convergence behavior of the conventional LMS algorithm. As noted in Introduction, rapid convergence results

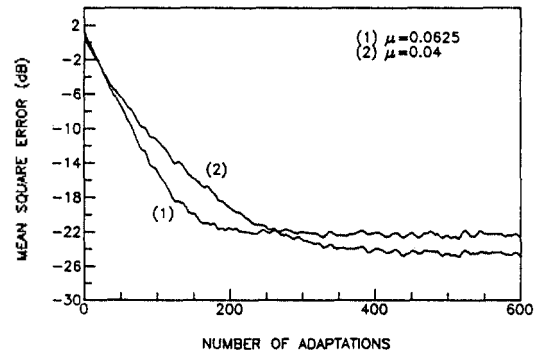


Fig. 2. Convergence characteristics of the LMS algorithm

in large steady state misadjustment and slow convergence small steady state misadjustment. Fig. 3. shows learning curves of the LMS algorithm with $\mu=0.0625$ and the SLMS algorithm with $\mu=0.0625$ and $\alpha=0.1$ 0.3 and, 0.5. As analyzed previously, we can see from this figure that convergence speeds of both algorithms are almost identical regardless of several values of α , but the misadjustment of the SLMS algorithm is smaller.

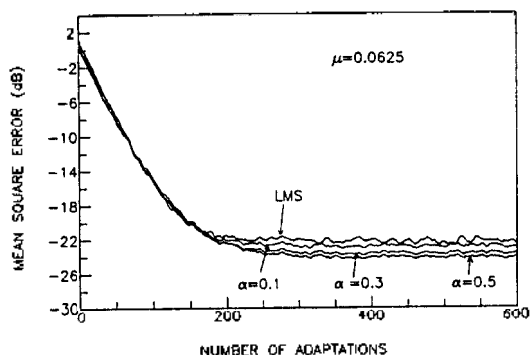


Fig. 3. Comparison of the convergence characteristics of the LMS and SLMS algorithms

Fig. 4. shows that the SLMS algorithm is more stable than the LMS algorithm when both algorithm have the same convergence factor. This is consistent with the fact that the convergence bound of the SLMS algorithm is increased, as analyzed in (21). Next, we discuss a

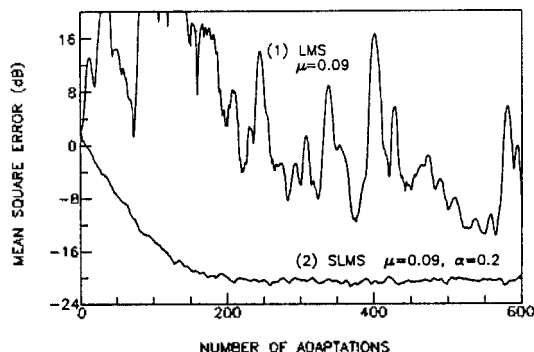


Fig. 4. Comparison of the convergence characteristics of the LMS and SLMS algorithms when the convergence factor is near the convergence bound

nonstationary case. Since the mathematical analysis of the nonstationary characteristic of an adaptive algorithm is very difficult, we investigate it by computer simulation. As for the LMS algorithm, a small convergence factor yields poor nonstationary performance[3]. This nonstationary performance of the LMS algorithm is shown in Fig. 5. The nonstationary situation for computer simulation is as follows. The channel impulse response, h_i , is assumed to be time-varying. For this time-varying impulse response, we introduce a time-varying W_n instead of a fixed W in (48) as follows.

$$w_n = 0.275 \sin\left(\frac{2n\pi}{400}\right) + 3.1$$

To obtain optimal weights corresponding to time-varying W_n , for every W_n , $n=1,2,\dots,400$, we carried out the LMS algorithm where μ was extremely small and the number of adaptations was very large. Fig. 5. shows the change of the 8-th component of the mean weight vector $E[w_n]$ over time under the nonstationary environment. In Fig. 5. as mentioned above, it can be observed that the weight vector of the system can track the optimal weight vector better with a larger μ

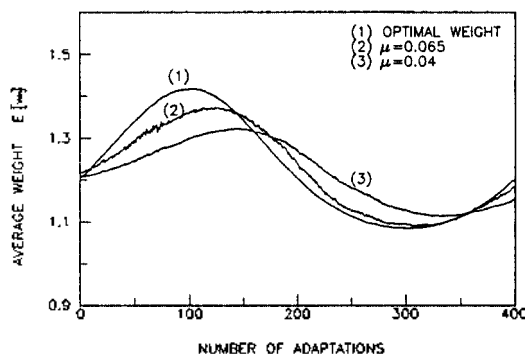


Fig. 5. Nonstationary characteristics of the LMS algorithm

Fig. 6. shows the nonstationary characteristics of the SLMS algorithm. As α becomes large, tracking capability of the SLMS algorithm becomes poor, which may be expected by intuition. Nevertheless, the tracking curves of the SLMS algorithm are similar to that of the LMS algorithm. In addition, Fig. 7. shows that when μ

is set to be a large value for better tracking, the SLMS algorithm keeps more stable track, while the LMS algorithm is locally unstable.

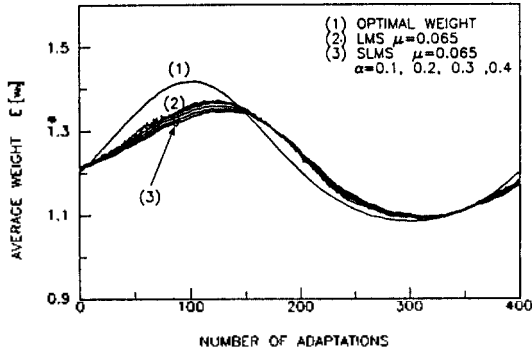


Fig. 6. Comparison of the nonstationary characteristics of the LMS and SLMS algorithms

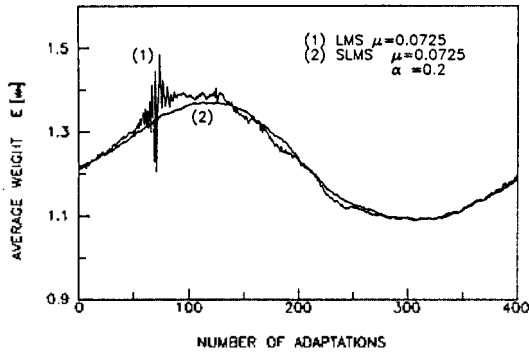


Fig. 7. Comparison of the nonstationary characteristics of the LMS and SLMS algorithms when the convergence factor is near the convergence bound

Now, we consider the ALE problem. Fig 8. shows the block diagram of ALE. The input to the ALE is given as

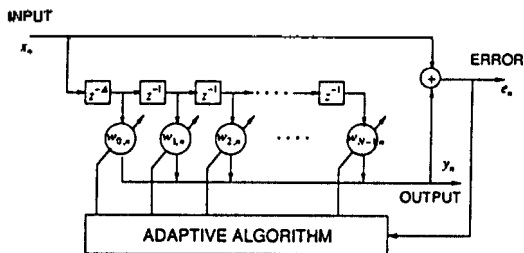


Fig. 8. Block diagram of ALE

$$x_n = \sqrt{2} \sin\left[\frac{2\pi k}{20}\right] + \sqrt{12} r_n \quad (49)$$

In (49), r_n has uniform distribution between -0.5 and 0.5, and therefore the signal-to-noise ratio (SNR) is unity, Fig 9. shows the output of ALE's using the LMS and SLMS algorithms. In Fig 9. the vertical line indicates the output of an ideal ALE, that is, spectral line with normalized frequency, $f = 0.05$. From Fig 9. we can see that the ALE using the SLMS algorithm estimates more accurate frequency and reduces more wide-band noise.

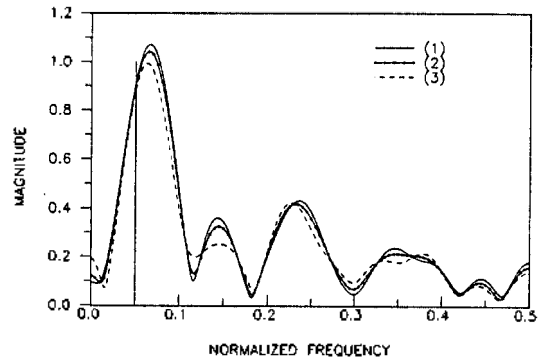


Fig. 9. Comparison of the frequency responses of adaptive filters in ALE using the LMS and SLMS algorithms ($\mu = 0.006$, $\text{SNR} = 1$)
(1) LMS (2) SLMS($\alpha=0.2$) (3) SLMS($\alpha=0.4$)

V. Conclusions

In this paper, we investigate the asymptotic performance of the SLMS algorithm and its applications. By re-formulating and analyzing the SLMS algorithm by Berman and Feuer, more useful results for understanding the algorithm is obtained. Since these results have useful forms in comparing with other LMS-type algorithms, they give some new perspectives on the performance of the SLMS algorithms. Especially, it is clearly shown that the convergence speed of the SLMS algorithm is asymptotically identical to that of the LMS algorithm. Furthermore, the SLMS al-

gorithm is compared with similar algorithms-the SGD and LFG algorithms, and the relationship on these algorithms is considered. In comparing with the SGD algorithm, we obtain the condition under which these two algorithms have the same steady state performance. In case of the LFG algorithm, the formulation of the algorithm is similar to that of the SLMS algorithm, but the convergence characteristics of the two algorithms are very different. We clearly explain this conflicting phenomenon, where we also show that the LFG and MLMS algorithms are exactly the same algorithm.

The analysis performed in this paper is verified by computer simulation. According to the computer simulation for a channel equalization, one can see that the SLMS algorithm is more stable than the LMS algorithm when a convergence factor is near the convergence bound. In addition, we investigate the nonstationary characteristics of the SLMS algorithm. In case of an ALE problem, we show that the ALE using the SLMS algorithm estimates more accurate frequency and reduces more wide-band noise.

APPENDIX

RELATIONSHIP BETWEEN ROOTS OF (19) AND α

Whether the roots of the equation (19) are real or complex depends on a value of α , and influences the convergence characteristics of the algorithm. In case that (19) has real roots, the following inequality should be satisfied :

$$\{\mu\lambda_i(1-\alpha) - (1+\alpha)\}^2 - 4\alpha \geq 0 \quad (A1)$$

After some manipulation, we may rewrite (A1) as

$$(\mu\lambda_i)^2 - 2\left[\frac{1+\alpha}{1-\alpha}\right]\mu\lambda_i + 1 \geq 0 \quad (A2)$$

From (A2),

$$0 \leq \mu\lambda_i \leq \frac{1+\alpha}{1-\alpha} - \sqrt{\left[\frac{1+\alpha}{1-\alpha}\right]^2 - 1} \quad (A3a)$$

or

$$\frac{1+\alpha}{1-\alpha} - \sqrt{\left[\frac{1+\alpha}{1-\alpha}\right]^2 - 1} \leq \mu\lambda_i < 2\left(\frac{1+\alpha}{1-\alpha}\right) \quad (A3b)$$

Since $\mu\lambda_i$ is small value, we are interested in (A3a). In (A3a), the smaller $\mu\lambda_i$ is, the larger value α may be allowed. For instance, if $\mu\lambda_i = 0.2$, the bound of α to allow the equation to have real roots is 0.443 and $\mu\lambda_i = 0.1$ the bound is 0.667. In practical circumstances, $\mu\lambda_i \ll 1$, and the equation, therefore, has real roots over a wide range of α .

REFERENCE

1. B. Widrow and M. E. Hoff, Jr., "Adaptive switching circuits," in 1960 WESCON Conv. Rec., pt.4, pp.96-140
2. B. Widrow and S. D. Stearns, *Adaptive signal processing*, Englewood Cliff, NJ : Prentice-Hall, 1985.
3. B. Widrow et al., "Stationary and nonstationary learning characteristics of the LMS adaptive filter," Proc. IEEE, vol. 64, pp.1151-1162, Aug. 1976.
4. J. R. Zeidler et al., "Adaptive enhancement of multiple sinusoids in uncorrelated noise," IEEE Trans. Acoust., Speech, Signal Processing, vol. ASSP-26, pp.240-254, June, 1978.
5. B. Widrow et al., "Adaptive noise cancelling : Principles and applications," Proc. IEEE, vol. 63, pp.1692-1716, Dec. 1975.
6. A. Gersho, "Adaptive equalization of highly dispersive channels for data transmission," Bell Syst. Tech. J., vol. 48, pp.55-70, Jan. 1969.
7. W. F. Gabriel, "Adaptive arrays-An introduction," Proce. IEEE, vol. 64, pp.239-272, Feb. 1976.
8. R. D. Gitlin and F. R. Magee, Jr., "Self-orthogonalizing adaptive equalization algorithms," IEEE Trans. Commun., vol. COM 25, pp.666-672, July, 1977.
9. S. Shaffer and C. S. Williams, "Comparison of

- the LMS, α -LMS, and data reusing LMS algorithms," in Proc. Seventeenth Asilomar Conf. on Circuits, Systems and Computers, Pacific Grove, CA., pp. 260-264, Nov. 1983.
10. A. Feuer and E. Weinstein, "Convergence analysis of LMS filters with uncorrelated Gaussian data," IEEE Trans. Acoust., Speech, Signal Processing, vol. ASSP-33, 222-230, Feb. 1985.
 11. S. Karni and G. Zeng, "A new convergence factor for adaptive filters," IEEE Trans. Circuit and Syst., vol. CAS-36, pp. 1011-1012, July 1989.
 12. A. Gardner, "Learning characteristics of stochastic gradient descent algorithms: A general study, analysis and critique," Signal Processing vol. 6, pp. 113-133, 1984.
 13. A. Feuer and N. Berman, "Performance analysis of the smoothed least mean square (SLMS) algorithm," Signal Processing, vol. 11, pp. 265-276, 1986.
 14. John g. Proakis, *Advances in Equalization of Intersymbol Interference, In Advances in Communication System*, vol. 4, A. J. Viterbi (ed.), New York: Academic Press, 1975.
 15. Ikjoo Chung, souguil Ann, "On the Asymptotic analysis of the smoothed LMS algorithm and the Relation with other LMS-type Algorithm," IEEE Tran. Circuit and Syst., Vol. 38, pp. 1551-1554, Dec. 1991.
 16. L. L. Horowitz and K. D. Senne, "Performance advantage of complex LMS for controlling narrowband adaptive arrays," IEEE Trans. Circuits Syst., vol. CAS-28, pp. 722-735, June 1981.
 17. N. S. Jayant and P. Noll, *Digital coding of Waveforms*. Englewood Cliffs, NJ: Prentice-Hall, 1984.
 18. J. J. Shynk and S. Roy, "The LMS algorithm with momentum updating," in Proc. Int. Symp. Circuits Syst., Espoo, Finland, June, 1988, pp. 2651-2654.
 19. J. C. Lee and C. K. Un, "Performance of transform-domain LMS adaptive digital filters," IEEE Trans. Acoust., Speech, Signal Processing, vol. ASSP-34, pp. 499-510, June, 1986.

▲Ikjoo Chung



Ikjoo Chung was born in Seoul, on January 9, 1964. He received the B.S., M.S., and Ph.D degrees in electronics engineering from Seoul National University in 1986, 1988, and 1992, respectively. From

1991 to 1992, he was a Member of Technical Staff at Dooin Electronic Co. in Seoul. Currently, he is an assistant professor in the department of Electronics at Kangwon National University. His research interest is adaptive signal processing, speech signal processing, and their applications.