

The performance of the hybrid LMS adaptive algorithm

Shiunn-Jang Chern^{a,*}, Jyh-Chau Horng^a, K. Max. Wong^b

^a *Institute of Electrical Engineering, National Sun Yat-Sen University, Kaohsiung 80424, Taiwan, ROC*

^b *Department of Electrical and Computer Engineering and Communication Research Laboratory, McMaster University, Ont., Hamilton, Canada*

Received 28 June 1993; revised 21 January 1994 and 21 June 1994

Abstract

The hybrid least mean square (HLMS) adaptive filter is a filter with an adaptation algorithm that is a combination of the conventional LMS algorithm and the normalized LMS (NLMS) algorithm. In this paper, the performance of the HLMS adaptive filtering algorithm is investigated. To do so, an analytical expression, in terms of the transient mean square error (MSE), is derived with application to the adaptive line enhancer (ALE). Based on this expression, we are able to examine the convergence properties of the HLMS. Simulation data using the ALE as an application verifies the accuracy of the analytical results. The performance of the HLMS algorithm is also compared with the conventional LMS algorithm as well as the NLMS algorithm. From the simulation results, we observed that, in general, the HLMS algorithm performs more robustly than the conventional LMS and the NLMS algorithms. Since the HLMS algorithm is a combination of the LMS algorithm and the NLMS algorithm, the selection of the optimum switching point of the HLMS algorithm is also addressed using a numerical approach. Many interesting characteristics of the switching point are obtained which show the relationship with the relevant parameters of the HLMS adaptive filter. The sensitivity of the selection of switching point is also examined.

Zusammenfassung

Das hybride least-mean-square (HLMS) adaptive Filter ist ein Filter mit einem Adaptionsalgorithmus, der eine Kombination aus dem konventionellen LMS-Algorithmus und dem normalisierten LMS-(NLMS)-Algorithmus darstellt. In dieser Arbeit werden die Eigenschaften des HLMS adaptiven Filteralgorithmus untersucht. Hierzu wird ein analytischer Ausdruck in Abhängigkeit des transienten mittleren quadratischen Fehlers (MSE) abgeleitet und auf einen adaptiven Leitungsentzerrer (ALE) angewendet. Basierend auf diesem Zusammenhang können wir die Konvergenzeigenschaften des HLMS untersuchen. Simulationsergebnisse unter Anwendung des ALE zeigen die Genauigkeit des analytischen Ergebnisses. Die Eigenschaften des HLMS-Algorithmus werden weiterhin mit dem konventionellen LMS-Algorithmus sowie mit dem NLMS-Algorithmus verglichen. Anhand der Simulationsergebnisse sehen wir, daß sich der HLMS-Algorithmus im allgemeinen robuster verhält als der konventionelle LMS und der NLMS-Algorithmus. Da der HLMS-Algorithmus eine Kombination des LMS- und des NLMS-Algorithmus darstellt, wird die Wahl des optimalen Umschaltzeitpunktes des HLMS anhand einer numerischen Lösung betrachtet. Man erhält viele interessante Eigenschaften des Umschaltzeitpunktes, die die Beziehungen mit den relevanten Parametern des HLMS adaptiven Filters aufzeigen. Es wird weiterhin die Empfindlichkeit bezüglich der Wahl des Umschaltzeitpunktes untersucht.

* Corresponding author.

Résumé

Le filtre adaptatif LMS hybride (HLMS) est un filtre dont l'algorithme d'adaptation est une combinaison de l'algorithme LMS conventionnel et de l'algorithme LMS normalisé (NLMS). Nous étudions dans cet article les performances de l'algorithme de filtrage adaptatif HLMS. Pour ce faire, une expression analytique en termes de l'erreur quadratique moyenne (MSE) transitoire est dérivée et appliquée au rehausseur de raie spectrale (ALE). Sur la base de cette expression, nous sommes capables d'examiner les propriétés de convergence du HLMS. Des données de simulation utilisant l'ALE comme application permettent de vérifier la précision des résultats analytiques. Les performances du HLMS sont également comparées à celles de l'algorithme LMS conventionnel ainsi qu'à celles de l'algorithme NLMS. A partir de ces résultats de simulation nous avons observé que, en général, l'algorithme HLMS se comporte de manière plus robuste que les algorithmes LMS et NLMS. Du fait que l'algorithme HLMS est une combinaison des algorithmes LMS et NLMS, la sélection du point optimal de commutation de l'algorithme HLMS est également étudiée à l'aide d'une approche numérique. De nombreuses caractéristiques intéressantes du point de commutation sont obtenues, explicitant la relation entre les paramètres significatifs du filtre adaptatif HLMS. La sensibilité de la sélection du point de commutation est également examinée.

Keywords: Normalized algorithm; Hybrid algorithm; Adaptive line-enhancer; Switching point; Convergence property

List of unusual symbols:

$\mathbf{x}(k)$	input signal vector
$z_i(k)$	the i th element of transformed domain signal vector
$n(k)$	noise signal
\mathbf{R}	autocorrelation matrix of $\mathbf{x}(k)$
$\text{Tr}\{\cdot\}$	trace of the matrix within the brackets
\mathbf{Q}	orthogonal transformation matrix
\mathbf{A}	diagonal matrix with entries being λ_i
$d(k)$	desired signal response
$y(k)$	adaptive filtered output
$s_i(k)$	the i th sinusoidal signal of $x(k)$
$e(k)$	error signal
$\xi(k)$	mean square error (of $e(k)$)
ξ_{\min}	minimum mean square error
$\mathbf{w}(k)$	time domain weight vector
\mathbf{w}_{opt}	optimum weight vector of $\mathbf{w}(k)$
$b_i(k)$	the i th element of transformed domain weight vector
\mathbf{p}	cross-correlation vector
N	filter length
L	number of sinusoidal signal
σ_s^2	variance of signals $s(k)$
σ_n^2	variance of noise $n(k)$
σ_d^2	variance of $d(k)$
$M_{\mathbf{w}}(k)$	mean weight vector of $\mathbf{w}(k)$
λ_i	the i th eigenvalue of \mathbf{R}
\mathbf{q}_i	the corresponding eigenvector of λ_i
$\text{Cov}[\mathbf{w}(k)]$	covariance matrix of $\mathbf{w}(k)$
$\Gamma(k)$	diagonal matrix with entries being eigenvalues of $\text{Cov}[\mathbf{w}(k)]$
γ_i	the i th eigenvalue of $\text{Cov}[\mathbf{w}(k)]$

$\mu(i)$	the step-size of the i th updated equation
μ_1	step-size for initial adaptation process
μ_2	step-size after initial adaptation process
η_0	scalar variable

1. Introduction

In various engineering applications, adaptive filtering techniques are very useful. This is due to the fact that they perform remarkably well over a wide range of input signal parameter statistics with no a priori information about the precise nature of these parameters. Therefore, in real-time applications they are very attractive.

Various adaptive filtering algorithms have been suggested by many researchers [9, 11, 13–15, 18]. The most commonly used is the least mean square (LMS) algorithm because of its simplicity, economy in computation, and ease in implementation. However, in the LMS adaptation algorithm the modes of the adaptive process converge at different rates such that the rate of each mode is determined by the associated eigenvalues of the reference autocorrelation matrix. For a large disparity of eigenvalues, in order to keep the algorithms stable, the LMS adaptive process may converge slowly. Under such circumstances, other algorithms with fast convergence properties are used.

The normalized LMS (NLMS) algorithm in the transform domain presented by Narayan et al. [13] is one of the methods that can achieve better convergence properties. Indeed, the equivalent time-domain expression of the NLMS algorithm in the transform domain has a form very similar to the recursive least square (RLS) algorithm, except for the constant step-size.

The implementation of the NLMS algorithm in the transform domain acquires the information of the power of the transformed input signal. However, because of the imperfection in the estimate of the power, the adaptation process of the NLMS algorithm may introduce an overshoot in the initial adaptation process. To solve the problem a hybrid LMS (HLMS) algorithm implemented in the transform domain adaptive filter is suggested here. The concept behind the HLMS algorithms is to use the merits of both the LMS and the NLMS algorithms in the transform domain to achieve the desired performance.

This paper first presents the HLMS algorithm and discusses the rationale behind it. Then the attention is focused on the analysis of the performance of the HLMS algorithm used in adaptive filtering. For the purpose of analysis, an equivalent time-domain expression of the transform domain HLMS adaptive algorithm is analyzed, and the

adaptive line-enhancer (ALE) is used as an application. The main result of this paper is that an analytical expression in terms of mean square error (MSE) is obtained. Based on this expression we are able to investigate the convergence property and show how it is related to the relevant parameters of the HLMS algorithm.

2. The hybrid LMS (HLMS) algorithm

The hybrid LMS (HLMS) adaptive filter, the adaptation algorithm of which is a combination of the LMS algorithm and the NLMS algorithm in the transform domain, is depicted in Fig. 1. Basically, the structure of the HLMS adaptive filter is very similar to that of the NLMS adaptive filter. The difference between both adaptive filters lies in the algorithms used for the adaptation process.

Without loss of generality, in the following discussion, the input signal is assumed to be a complex-valued random process. From Fig. 1, the equation of up-dating the weights of the HLMS algorithm can be expressed as

$$b_i(k + 1) = b_i(k) + 2\mu(i)e(k)z_i^*(k), \tag{1a}$$

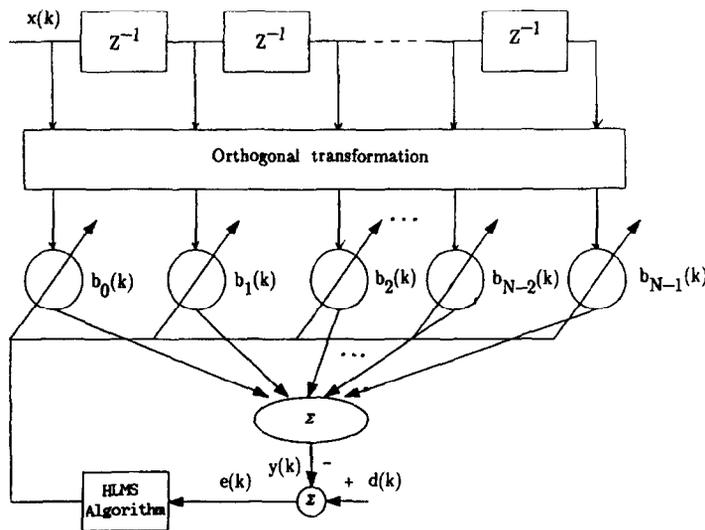


Fig. 1. The transformed domain HLMS adaptive filter.

with

$$\mu(i) = \begin{cases} \mu_1; & \text{for initial adaptation process,} \\ \frac{\mu_2}{E[|z_i(k)|^2]}; & \text{when input signal samples have} \\ & \text{been accumulated sufficiently,} \end{cases} \quad (1b)$$

$$i = 0, 1, \dots, N - 1,$$

where * denotes complex conjugate and both μ_1 and μ_2 are the step-sizes which control the speed of convergence and stability. In Eq. (1a), $b_i(k)$ is the i th component of the transformed tap-weight at k th time instant and $z_i(k)$ is the corresponding transformed input signal. Accordingly, the expected value of $|z_i(k)|^2$, $E[|z_i(k)|^2]$, represents the average power of $z_i(k)$. Also, in Eq. (1a), $e(k)$ is the error signal which is defined as the difference of the desired signal, $d(k)$, from the filtered signal, $y(k)$, i.e., $e(k) = d(k) - y(k)$.

As can be seen from Eq. (1b), if the value of $\mu(i)$ is taken to be $\mu_2/E[|z_i(k)|^2]$ in the entire adaptation process then we have the NLMS algorithm. On the other hand, if only μ_1 is used for the adaptation, then it will be the LMS algorithm in the transform domain. As addressed in [13] the LMS algorithm under an orthogonal transform domain can be shown to be equivalent to the conventional time-domain LMS algorithm. From this point of view, the HLMS algorithm can be visualized as a generalized expression of both the LMS and NLMS algorithms in the transform domain depending on how the value of $\mu(i)$ is selected.

Because of insufficient input signals used to estimate the average power of the input signal, overshoot may occur in the NLMS algorithm [2]. Therefore, we propose in Eq. (1b) that at the beginning of the adaptation process, $\mu(i)$ is chosen to be a constant value, μ_1 . As described in [6], if we expect the LMS algorithm to have a faster convergence rate, μ_1 should be assigned a value which will be near $1/3 \text{Tr}\{\mathbf{R}\}$. The time-domain input autocorrelation matrix, \mathbf{R} , is defined as $\mathbf{R} = E[\mathbf{x}^*(k) \mathbf{x}^T(k)]$ and $\text{Tr}\{\cdot\}$ denotes the trace of the matrix within brackets. Here $\mathbf{x}(k)$ is the input signal vector and superscript T denotes the transpose operator. Once we have accumulated sufficient signal samples for

estimating the power of $z_i(k)$ such that the overshoot can be eliminated, $\mu(i)$ is then changed to the value of $\mu_2/E[|z_i(k)|^2]$. Here μ_2 has to satisfy the constraint as in the NLMS algorithm [4, 12].

The equivalent time-domain expression of Eq. (1) can be easily shown to be [13, 15]

$$\mathbf{w}(k+1) = \mathbf{w}(k) + 2\mu_1 e(k) \mathbf{x}^*(k) \quad \text{for } 0 < k \leq p \quad (2a)$$

and

$$\mathbf{w}(k+1) = \mathbf{w}(k) + 2\mu_2 \mathbf{R}^{-1} e(k) \mathbf{x}^*(k) \quad \text{for } k \geq p+1 \quad (2b)$$

with integer p be the switching point, provided that the orthogonal transformation matrix, in Fig. 1, completely diagonalized \mathbf{R} . Here \mathbf{R}^{-1} is the inverse matrix of \mathbf{R} and p is the point in which we turn the HLMS algorithm from the LMS algorithm mode into the NLMS algorithm mode. The problem of selecting the switching point will be investigated in Section 4.1. Also, in Eqs. (2), $\mathbf{w}(k)$ and $\mathbf{x}(k)$ are the time-domain weight vector and the input signal vector at time instant k , respectively, and are defined as $\mathbf{w}(k) = [w_1(k) \ w_2(k) \ \dots \ w_N(k)]^T$ and $\mathbf{x}(k) = [x(k) \ x(k-1) \ \dots \ x(k-N+1)]^T$. For simplicity, our analysis will be based on Eq. (2) instead of Eq. (1).

3. Performance of the HLMS algorithm used as an ALE

In this section, we apply the HLMS algorithm to an adaptive line-enhanced (ALE). The performance of the ALE is then analyzed.

The ALE is an adaptive filter configuration used for the detection of a narrow-band signal in the presence of broad-band noise. It was first proposed by Widrow et al. [19] and extensively discussed by many others [5, 7, 21]. The ALE becomes a competitor of the fast Fourier algorithm as a sensitive detector and has capabilities that may exceed those of conventional spectral analyzers when the unknown sinusoidal wave has finite bandwidth or is frequency modulated [8, 20]. Together with the adaptive notch filter, the ALE can also be used as

a pre-processor to further improve the resolution of frequency estimation for short data records [3].

As shown in Fig. 2, the ALE is implemented with a two-channel processor in which a delayed version of the received signal is adaptively filtered and subtracted from the instantaneous received signal. It has been shown to be an adaptive implementation of a Δ -step Wiener predictor [21]. Here the delay parameter Δ is called the decorrelation parameter of the ALE. The main function of delay parameter Δ is to remove correlation that may exist between the broad-band components of the two channels and introducing a simple phase shift between the narrow-band signals. Therefore, the ALE can be viewed as a signal separator, i.e., it separates the desired component from the undesired component.

In order to carry out the theoretical analysis of the ALE, in terms of MSE, certain assumptions for the input signals have to be made. Fisher and Bershad [7] utilized the complex time-domain LMS adaptation algorithm to theoretically investigate the behavior of the ALE. The signal model assumed by Fisher and Bershad [7] included multiple sinusoids in broad-band noise. It is also assumed that each sinusoid signal is narrow band, statistically independent, orthogonal, and Rayleigh

fading with equal power. Here, the signal model used in [7] was extended by Chern [1] to the case of multiple sinusoids with unequal power. This extended signal model is available for investigating the effect on convergence rate due to the unequal power of the complex LMS ALE. For the purpose of comparison, again, the extended signal model is considered for the present analysis.

3.1. Signal model description

Consider an ALE with length N which is used to estimate L statistically independent, orthogonal, stationary, and relatively slowly varying Rayleigh fading complex sinusoids in additive circular Gaussian while noise [16] independent of the sinusoids. The i th complex sinusoid can be written in the form, e.g., $s_i(k) = A_i(k) \exp(j\omega_i k T_s)$, $k = 0, 1, 2, \dots$, with $j = \sqrt{-1}$. Here ω_i represents the i th carrier frequency and T_s is the sampling interval. Also, $A_i(k)$ denotes a slowly varying, zero-mean circular normal Gaussian [16] complex valued envelope of the i th sinusoid with the following properties, e.g., $E\{A_i(k)A_m^*(n)\} = 0$, $i \neq m$ and $E\{A_i(k)A_i^*(n)\} = 0$, if $|k - n| > N$, or $E\{A_i(k)A_i^*(n)\} = \sigma_s^2$, if

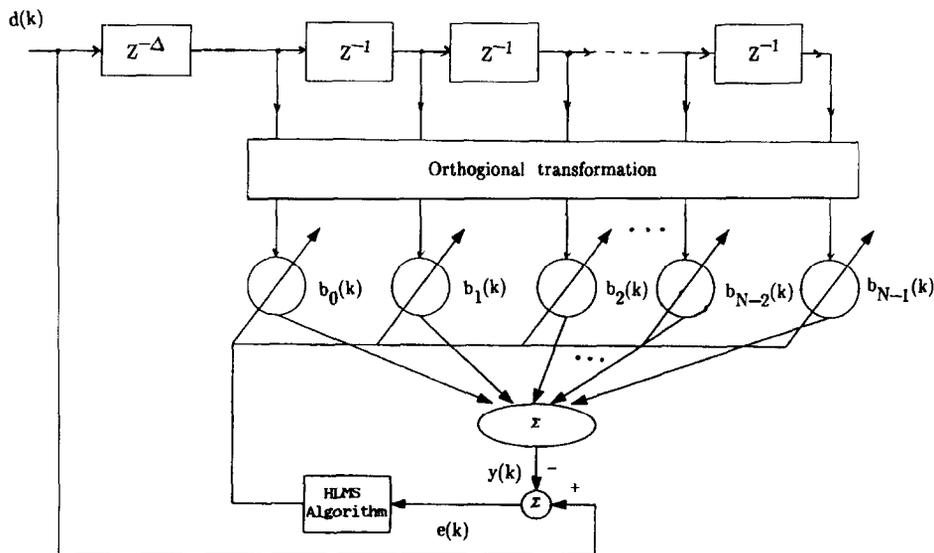


Fig. 2. The transformed domain HLMS adaptive line-enhancer.

$|k - n| \leq N$, for $i = 1, 2, \dots, L$. Here $\sigma_{s_i}^2$ is the variance of the i th sinusoidal signal. This will consist of the general Gaussian assumption described earlier. Moreover, the additive circular Gaussian white noise, $n(k)$, is assumed to be zero mean with the statistical property $E\{n(k)n^*(i)\} = \sigma_n^2$, if $k = i$, or $E\{n(k)n^*(i)\} = 0$, if $k \neq i$. Here σ_n^2 is the variance of $n(k)$. The assumptions described for above model are such that the ALE operates on narrow-band Gaussian sinusoidal inputs with small bandwidth compared to the reciprocal of the filter length, N .

Now, let $d(k)$ denote the received signal of the ALE. Since the input of the ALE is the delayed version of received signal $d(k)$, consequently, the input autocorrelation matrix, \mathbf{R} , can be expressed as

$$\mathbf{R} = \sigma_n^2 \mathbf{I} + \sum_{i=1}^L \sigma_{s_i}^2 \mathbf{v}_i \mathbf{v}_i^H, \quad (4)$$

where \mathbf{I} is the identity matrix and the superscript H denotes the complex conjugate transpose operator. In Eq. (4), \mathbf{v}_i , $i = 1, 2, \dots, L$, denote the vectors

$$\mathbf{v}_i = [1 e^{j\omega_i T_s} \dots e^{j(N-1)\omega_i T_s}]^T, \quad i = 1, 2, \dots, L. \quad (5)$$

Under the assumption that $\{\mathbf{v}_i\}$ is a set of orthogonal vectors, the eigenvalues of \mathbf{R} are given as

$$\lambda_i = \sigma_n^2 + N\sigma_{s_i}^2, \quad i = 1, 2, \dots, L, \quad (6a)$$

with associated eigenvectors

$$\mathbf{q}_i = \mathbf{v}_i / \sqrt{N}, \quad i = 1, 2, \dots, L \quad (6b)$$

and

$$\lambda_i = \sigma_n^2, \quad i = L + 1, L + 2, \dots, N, \quad (6c)$$

with associated eigenvectors \mathbf{q}_i , $i = L + 1, L + 2, \dots, N$, which form a set of orthonormal vectors and are orthogonal to \mathbf{q}_i , $i = 1, 2, \dots, L$.

Similarly, the cross-correlation vector of $d(k)$ and $\mathbf{x}(k)$ is given by

$$\mathbf{p} = E[d(k)\mathbf{x}^*(k)] = \sum_{i=1}^L \beta_i \sigma_{s_i}^2 \mathbf{v}_i, \quad (7)$$

where β_i , $i = 1, 2, \dots, L$, are complex phase factors depending on the lag between $\mathbf{x}(k)$ and $d(k)$.

3.2. Mean weight vector of the ALE

Basically, the analysis of the HLMS algorithm is carried out in a similar manner as in [1, 7]. To begin the analysis, we start with $\mathbf{w}(0)$, the initial weight vector, and iteratively substituting in Eqs. (2), results in [7]

$$\begin{aligned} \mathbf{w}(k) = & \prod_{i=0}^{k-1} [\mathbf{I} - 2\mu_i \mathbf{x}^*(i)\mathbf{x}^T(i)] \mathbf{w}(0) \\ & + 2\mu_1 d(k-1)\mathbf{x}^*(k-1) \\ & + 2\mu_1 \sum_{i=0}^{k-2} \prod_{m=i+1}^{k-1} [\mathbf{I} - 2\mu_1 \mathbf{x}^*(m)\mathbf{x}^T(m)] \\ & \quad \times d(i)\mathbf{x}^*(i) \quad \text{for } 0 < k \leq p. \end{aligned} \quad (8a)$$

It is noted that Eq. (8a) is in the conventional time-domain LMS algorithm mode. Since p is the switching point, for k equal or greater than $p + 1$ the weight update equation will be in the NLMS algorithm mode. Similarly, if we start with $\mathbf{w}(p + 1)$, the initial weight vector in the mode of NLMS algorithm, the HLMS algorithm weight update equation can be obtained as

$$\begin{aligned} \mathbf{w}(k) = & \prod_{i=p}^{k-1} [\mathbf{I} - 2\mu_2 \mathbf{R}^{-1} \mathbf{x}^*(p)\mathbf{x}^T(p)] \\ & \times \left\{ \prod_{i=0}^{p-1} [\mathbf{I} - 2\mu_1 \mathbf{x}^*(i)\mathbf{x}^T(i)] \mathbf{w}(0) \right. \\ & \quad + 2\mu_1 \sum_{i=0}^{p-2} \prod_{m=i+1}^{p-1} [\mathbf{I} - 2\mu_1 \mathbf{x}^*(m)\mathbf{x}^T(m)] \\ & \quad \left. \times d(i)\mathbf{x}^*(i) + 2\mu_1 d(p-1)\mathbf{x}^*(p-1) \right\} \\ & + 2\mu_2 \mathbf{R}^{-1} \sum_{i=p}^{k-2} \left\{ \prod_{m=i+1}^{k-1} [\mathbf{I} - \right. \\ & \quad \left. 2\mu_2 \mathbf{R}^{-1} \mathbf{x}^*(m)\mathbf{x}^T(m)] \right\} d(i)\mathbf{x}^*(i) \\ & + 2\mu_2 \mathbf{R}^{-1} d(k-1)\mathbf{x}^*(k-1) \end{aligned} \quad (8b)$$

for $k \geq p + 1$.

As discussed in [7] (see [7] after Eq. (4)), in Eq. (8) $\mathbf{w}(k)$ is a function of only $i < k$ elements of the sequences $\{d(i)\}$, $\{\mathbf{x}(i), \mathbf{x}^*(i)\}$. Thus, it is concluded that the elements of the vectors $\mathbf{w}(k)$ and $\mathbf{x}(k)$ are mutually statistically independent if $\mathbf{x}(k)$ is statistically independent of the $i \neq k$ elements of the sequences $\{d(i), d^*(i)\}$, $\{\mathbf{x}(i), \mathbf{x}^*(i)\}$. While this is a common assumption in adaptive filtering analysis (particularly when μ_1 is small enough). As addressed in [17] (see [17] after Eq. (5)), this is also an excellent assumption for the ALE even for highly correlated inputs and values of μ_1 approaching the stability limit of the LMS algorithm.

To derive the mean square error (MSE) in the transient state, we need to obtain the recursive mean weight vector and the weight covariance matrix. Under the assumption just described and following the similar approach as in [7], the corresponding mean weight vector of Eqs. (8) can be written as

$$M_w(k) = E[\mathbf{w}(k)] = [\mathbf{I} - 2\mu_1 \mathbf{R}]^k M_w(0) + 2\mu_1 \sum_{i=0}^{k-1} (\mathbf{I} - 2\mu_1 \mathbf{R})^i \mathbf{p} \quad \text{for } k \leq p, \quad (9a)$$

in the LMS algorithm mode and

$$M_w(k) = (1 - 2\mu_2)^{k-p} \left\{ 2\mu_1 \sum_{i=0}^{p-1} (\mathbf{I} - 2\mu_1 \mathbf{R})^i \mathbf{p} + (\mathbf{I} - 2\mu_1 \mathbf{R})^p M_w(0) \right\} + 2\mu_2 \mathbf{R}^{-1} \sum_{i=p}^{k-1} (1 - 2\mu_2)^{k-i-1} \mathbf{p} \quad \text{for } k \geq p + 1, \quad (9b)$$

in the NLMS algorithm mode, respectively. In Eqs. (9), $M_w(0)$ is the initial mean weight vector and its value can be arbitrarily selected. In steady-state, Eq. (9b) becomes

$$\lim_{k \rightarrow \infty} M_w(k) = \mathbf{w}_{\text{opt}} = \mathbf{R}^{-1} \mathbf{p}. \quad (10)$$

This is because in the steady state, the first term on the right-hand side of Eq. (9b) will vanish and the last term is reduced to Wiener weight vector, provided that $0 < \mu_2 \leq 1$. Now, applying the signal

model of ALE described in Section 3.1 to the mean weight vector of Eqs. (9), results in [7]

$$M_w(k) = \sum_{i=1}^L \beta_i \frac{\sigma_{s_i}^2}{\lambda_i} \{1 - [1 - 2\mu_1 \lambda_i]^k\} \mathbf{v}_i \quad \text{for } k \leq p \quad (11a)$$

and (from Appendix A)

$$M_w(k) = \sum_{i=1}^L \beta_i \frac{\sigma_{s_i}^2}{\lambda_i} \{1 - (1 - 2\mu_1 \lambda_i)^p\} (1 - 2\mu_2)^{k-p} \mathbf{v}_i \quad \text{for } k \geq p + 1, \quad (11b)$$

respectively, where for simplicity, $M_w(0)$ is assumed to be a null vector.

3.3. Weight covariance matrix of the ALE

To obtain the weight covariance matrix recursive equation, we can follow a similar procedure as in [7]. First, we consider the weight covariance matrix recursive equation of the HLMS algorithm in the LMS algorithm mode. Since it is nothing but an LMS algorithm (for $k \leq p$), we have [1]

$$\begin{aligned} \text{Cov}[\mathbf{w}(k+1)] &= E\{[\mathbf{w}(k+1) - E[\mathbf{w}(k+1)]] [\mathbf{w}(k+1) - E[\mathbf{w}(k+1)]]^H\} \\ &= [\mathbf{I} - 4\mu_1 \mathbf{R}] \text{Cov}[\mathbf{w}(k)] + 4\mu_1^2 \mathbf{R} \text{Cov}[\mathbf{w}(k)] \mathbf{R} \\ &\quad + 4\mu_1^2 \{ \text{Tr}[\mathbf{R} \text{Cov}[\mathbf{w}(k)]] - \mathbf{p}^H M_w(k) - M_w^H(k) \mathbf{p} + \sigma_d^2 + M_w^H(n) \mathbf{R} M_w(k) \} \mathbf{R} \end{aligned} \quad \text{for } k \leq p, \quad (12a)$$

where σ_d^2 is the variance of $d(k)$. On the other hand, the recursive equation in the NLMS algorithm mode can be derived (Appendix B)

$$\begin{aligned} \text{Cov}[\mathbf{w}(k+1)] &= (1 - 2\mu_2)^2 \text{Cov}[\mathbf{w}(k)] \\ &\quad + 4\mu_2^2 \text{Tr}\{\mathbf{R} \text{Cov}[\mathbf{w}(k)]\} \mathbf{R}^{-1} \\ &\quad + 4\mu_2^2 \mathbf{R}^{-1} \{ \sigma_d^2 + M_w^H(k) \mathbf{R} M_w(k) - \mathbf{p}^H M_w(k) - M_w^H(k) \mathbf{p} \} \quad \text{for } k \geq p + 1. \end{aligned} \quad (12b)$$

Now, to simplify the expression of the transient MSE it will be better to factorize the recursive weight covariance matrix equation, in terms of eigenvalues and the corresponding eigenvectors of $\text{Cov}[\mathbf{w}(k)]$. As in [7], we can similarly show that if the \mathbf{Q} matrix diagonalizes \mathbf{R} , it also diagonalizes the weight covariance matrix, provided that the initial weight covariance matrix is diagonalized by \mathbf{Q} . Here the matrix \mathbf{Q} is an $N \times N$ square matrix with its columns being the eigenvectors of \mathbf{R} . Thus the weight covariance matrix at time k can be expressed as

$$\text{Cov}[\mathbf{w}(k)] = \mathbf{Q}\mathbf{\Gamma}(k)\mathbf{Q}^H = \sum_{i=1}^N \gamma_i(k)\mathbf{q}_i\mathbf{q}_i^H, \quad (13)$$

where $\mathbf{\Gamma}(k)$ is a diagonal matrix with its entry elements being $\{\gamma_i(k); i = 1, 2, \dots, N\}$:

$$\mathbf{\Gamma}(k) = \text{diag}[\gamma_1(k) \ \gamma_2(k) \ \dots \ \gamma_N(k)]. \quad (14)$$

Consequently, pre- and post-multiplying Eq. (12a) and Eq. (12b) by \mathbf{Q}^H and \mathbf{Q} , respectively, we have

$$\begin{aligned} \mathbf{\Gamma}(k+1) &= \mathbf{\Gamma}(k) - 4\mu_1\mathbf{A}\mathbf{\Gamma}(k) + 4\mu_1^2\mathbf{A}\mathbf{\Gamma}(k)\mathbf{A} \\ &\quad + 4\mu_1^2 \text{Tr}\{\mathbf{A}\mathbf{\Gamma}(k)\}\mathbf{A} \\ &\quad + 4\mu_1^2\eta_0(k)\mathbf{A} \quad \text{for } k \leq p, \end{aligned} \quad (15a)$$

where $\eta_0(k)$ will be defined in Eq. (16) and

$$\begin{aligned} \mathbf{\Gamma}(k+1) &= (1 - 2\mu_2)^2\mathbf{\Gamma}(k) + 4\mu_2^2\eta_0(k)\mathbf{A}^{-1} \\ &\quad + 4\mu_2^2 \text{Tr}\{\mathbf{\Gamma}(k)\mathbf{A}\}\mathbf{A}^{-1} \\ &\quad \text{for } k \geq p+1, \end{aligned} \quad (15b)$$

where \mathbf{A} is a diagonal matrix with its entry being the eigenvalues of \mathbf{R} . In Eqs. (15), the scalar variable $\eta_0(k)$ is defined as

$$\eta_0(k) = \xi_{\min} + [\mathbf{M}_w(k) - \mathbf{w}_{\text{opt}}]^H \mathbf{R} [\mathbf{M}_w(k) - \mathbf{w}_{\text{opt}}] \quad (16)$$

and the minimum MSE is given by

$$\xi_{\min} = \sigma_d^2 - \mathbf{w}_{\text{opt}}^H \mathbf{p}. \quad (17)$$

3.4. Transient MSE of the ALE

To evaluate the performance the transient convergence property is of interest and usually

described by the learning curve of the transient MSE. The MSE, by definition, is given by $\xi(k) = E[|e(k)|]$. With $y(k) = \mathbf{x}^T(k)\mathbf{w}(k)$ and after some manipulation, we have

$$\begin{aligned} \xi(k) &= \eta_0(k) + \text{Tr}\{\mathbf{R} \text{Cov}[\mathbf{w}(k)]\} \\ &= \eta_0(k) + \sum_{i=1}^N \gamma_i(k)\lambda_i. \end{aligned} \quad (18)$$

From Eq. (18), we see that to evaluate the transient MSE, we simply compute the scalar variables $\eta_0(k)$ and $\gamma_i(k)$, $i = 1, 2, \dots, N$, at each time instant k . For convenience of analysis, here we consider the case of two unequal power sinusoidal signals (i.e., $L = 2$) buried in additive white noise. The reason of using the unequal power sinusoids for analysis was described earlier.

For simplicity, Eqs. (15a) and (15b) can be written with reduced rank; for $L = 2$, we have

$$\begin{aligned} \gamma_i(k+1) &= (1 - 4\mu_1\lambda_i + 4\mu_1^2\lambda_i^2)\gamma_i(k) \\ &\quad + 4\mu_1^2\lambda_i \sum_{j=1}^N \lambda_j\gamma_j(k) + 4\mu_1^2\lambda_i\eta_0(k), \\ &\quad i = 1, 2, 3, \quad \text{for } k \leq p \end{aligned} \quad (19a)$$

in the LMS algorithm mode and

$$\begin{aligned} \gamma_i(k+1) &= (1 - 4\mu_2 + 4\mu_2^2)\gamma_i(k) \\ &\quad + 4\mu_2^2\lambda_i^{-1} \sum_{j=1}^N \lambda_j\gamma_j(k) + 4\mu_2^2\lambda_i^{-1}\eta_0(k), \\ &\quad i = 1, 2, 3, \quad \text{for } k \geq p+1 \end{aligned} \quad (19b)$$

in the NLMS algorithm mode. Here $\gamma_i(k)$, $i = 1, 2, 3$, in Eqs. (19) are associated with the corresponding eigenvectors \mathbf{v}_i/\sqrt{N} , $i = 1, 2$ and \mathbf{q}_i , $i = 3$ (see Eqs. (6)), respectively. Under the assumption that the initial mean weight vector is a null vector and starting with $\gamma_i(0)$, then based on Eqs. (19), we can evaluate $\gamma_i(k)$, for all k . To evaluate the MSE, an explicit expression of $\eta_0(k)$, in Eq. (16), has to be derived. In Eq. (16), the error weight vector is defined as the difference of the mean weight vector and the optimum weight vector. Applying Eqs. (4), (7), (10) and (11) to the error weight vector, we

have

$$M_w(k) - w_{\text{opt}} = - \sum_{i=1}^2 \beta_i \frac{\sigma_{s_i}^2}{\lambda_i} (1 - 2\mu_1 \lambda_i)^k \mathbf{v}_i$$

for $k \leq p$ (20a)

and

$$M_w(k) - w_{\text{opt}} = - \sum_{i=1}^2 \beta_i \frac{\sigma_{s_i}^2}{\lambda_i} (1 - 2\mu_1 \lambda_i)^p (1 - 2\mu_2)^{k-p} \mathbf{v}_i$$

for $k \geq p + 1$, (20b)

respectively. Substituting Eqs. (20) into Eq. (16), accordingly, we have

$$\eta_0(k) = \xi_{\min} + \sum_{i=1}^2 (1 - 2\mu_1 \lambda_i)^{2k} \frac{(\sigma_{s_i}^2)^2}{\lambda_i} N$$

for $k \leq p$ (21a)

and

$$\eta_0(k) = \xi_{\min} + \sum_{i=1}^2 (1 - 2\mu_1 \lambda_i)^{2p} (1 - 2\mu_2)^{2(k-p)} \frac{(\sigma_{s_i}^2)^2}{\lambda_i} N$$

for $k \geq p + 1$. (21b)

In Eqs. (21), the value of ξ_{\min} can be easily evaluated from Eq. (17) by applying the corresponding parameters to it. For $L = 2$, based on Eqs. (6), (18), (19), and (21), the transient MSE for the HLMS algorithm can be evaluated. However, the transient MSE can be expressed in a more explicit form. This is derived in the following.

To have a closed-form expression of the transient MSE, first, we substitute Eq. (19b) into the second term on the right-hand side of Eq. (18), yields

$$\sum_{i=1}^N \lambda_i \gamma_i(k) = [1 - 4\mu_2 + (N + 1)4\mu_2^2] \sum_{i=1}^N \lambda_i \gamma_i(k-1) + 4\mu_2^2 N \eta_0(k-1)$$

$$= [1 - 4\mu_2 + (N + 1)4\mu_2^2]^{k-p} \sum_{i=1}^N \lambda_i \gamma_i(p) + \sum_{i=0}^{k-p-1} [1 - 4\mu_2 + (N + 1)4\mu_2^2]^i 4\mu_2^2 N \times \eta_0(k-i-1) \quad \text{for } k \geq p + 1. \quad (22)$$

Applying Eq. (22) to Eq. (18), the MSE in the NLMS algorithm mode can be expressed as

$$\xi(k) = (1 - 4\mu_2 + (N + 1)4\mu_2^2)^{k-p} \sum_{i=1}^N \lambda_i \gamma_i(p) + \sum_{i=0}^{k-p-1} [1 - 4\mu_2 + (N + 1)4\mu_2^2]^i \times 4\mu_2^2 N \eta_0(k-i-1) + \eta_0(k)$$

for $k \geq p + 1$. (23)

As observed from Eq. (23), to have the transient MSE in a closed form, we need the derivation of $\gamma_i(p)$, in terms of its initial value. However, since the values of $\gamma_i(p)$, $i = 1, 2, \dots, N$, are indeed in [1] (see Appendix C), can be directly applied. Thus, the closed-form expression of the transient MSE in the NLMS algorithm mode can be obtained. It should be pointed out, at this moment, that the performance evaluation of the NLMS algorithm, in terms of transient MSE, can be directly obtained using the HLMS algorithm with $p = 0$.

4. Numerical analysis of convergence rate and the choice of switching point

To evaluate the performance of the equivalent time-domain expression of the HLMS adaptive algorithm, a computer simulation is carried out for the analytical expression of MSE derived in last section. The results so obtained are also compared with the results of the LMS algorithm as well as the NLMS algorithm. Moreover, as described earlier, in the HLMS algorithm a switching point has to be determined, in which the LMS algorithm mode turns into the NLMS algorithm mode when the input signals are sufficient to accurately estimate the received signal power.

Thus, the objective of this section is to examine the convergence property as well as to determine the value of switching point in different environments. The problem of selecting the switching point will be discussed first. It is noted that, in the following computer simulation, the parameter SNR (the signal-to-noise ratio) is defined as the power ratio of the largest sinusoidal signal to the noise power. Also, SSR is defined as the power ratio between the largest sinusoidal and the smallest sinusoidal components in the received signal.

4.1. The choice of the switching point

In the HLMS adaptation algorithm, the problem of selecting the switching point is very important. This is because the HLMS algorithm is a combination of the LMS algorithm and the NLMS algorithm, therefore, the transient convergence rate is dominated by the value of p , the switching point. Based on the explicit expressions obtained in last section, we are able to find out the optimum value of the switching point in different situations.

The objective of this section is to answer the following question. In case of ALE, what is the proper value of the parameter, p , that yields the smallest MSE at a fixed observation time, n , when other parameters in HLMS algorithm are given.

The optimum switching point can be obtained by minimizing the MSE over p . To do so, one can differentiate Eq. (23) with respect to p , for $k > p$, and set the result to zero, having other parameters fixed. However, Eq. (23) is a complicated function which involves the computation of eigenvalues, eigenvectors, matrix inversion and matrix multiplication. Thus, an explicit solution is not easy to obtain. An alternative way using the numerical approach may be chosen to determine the switching point. That is, for different cases, with different values of p , we can figure out the minimum value of switching point from the individual learning curves, in terms of transient MSE. These results are shown in Figs. 3–7. We found that these graphs exhibit some interesting characteristics which are described as follows:

1. A common feature observed from Figs. 3 to 7 is that the larger is the tap-weight length, N , the

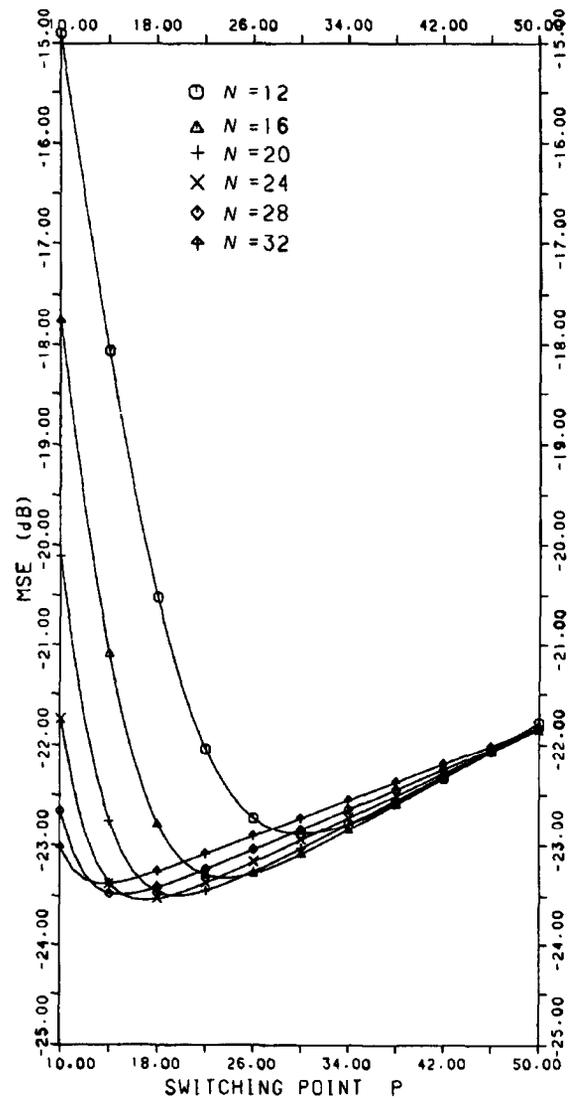


Fig. 3. The MSE versus switching point for varied weight length with SNR = 40 dB, SSR = 20 dB, $\mu_1 = 0.005$, $\mu_2 = 0.005$.

smaller is the optimum value of switching point. This is because for larger value N , as can be seen in Fig. 8, the ALE has a faster convergence rate in the initial adaptation process, yielding the result described above.

2. As shown in Figs. 3 and 4, for different values of SNR, with other parameters fixed, the optimum value of p will be approximately the same.
3. Let us see the effect of p due to the change of step-sizes, μ_1 and μ_2 . From Figs. 4–6, it can be

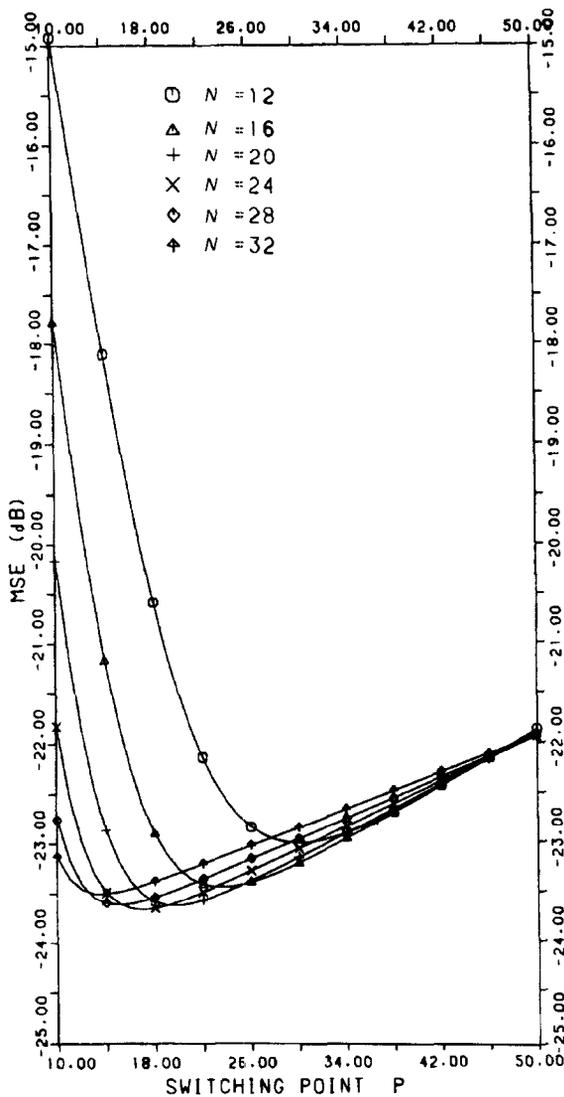


Fig. 4. The MSE versus switching point for varied weight length with SNR = 60 dB, SSR = 20 dB, $\mu_1 = 0.005$, $\mu_2 = 0.005$.

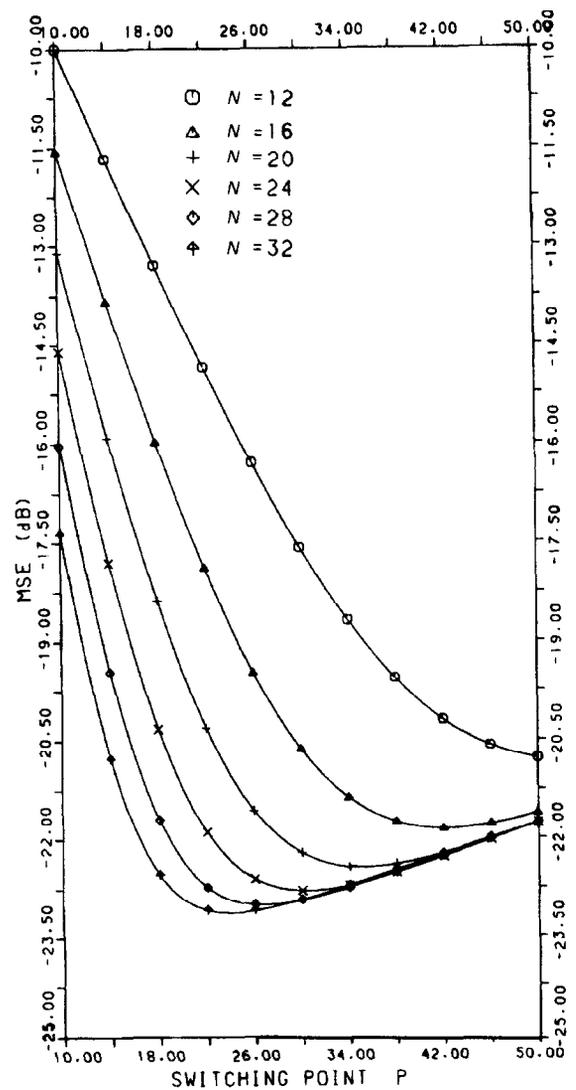


Fig. 5. The MSE versus switching point for varied weight length with SNR = 60 dB, SSR = 20 dB, $\mu_1 = 0.0025$, $\mu_2 = 0.005$.

seen that the value of optimum p increases as the step-sizes, μ_1 and μ_2 , decrease.

- Comparing Fig. 7 with Fig. 4, we can see that the value of optimum p with lower SSR is much larger than that with relatively higher SSR. The reason of this is due to the fact that in the LMS algorithm mode with SSR = 0 dB (equal power of sinusoidal signals), a fast convergence rate can be achieved, as shown in Fig. 8 [1].

4.2. Convergence property of the HLMS ALE

In this section, the convergence property of the HLMS algorithm for both theoretical results and simulation data is examined. To illustrate the convergence property, again, ALE is considered. Also, to verify the accuracy of the theoretical results, the received signal described in Eq. (24) is used to perform the computer

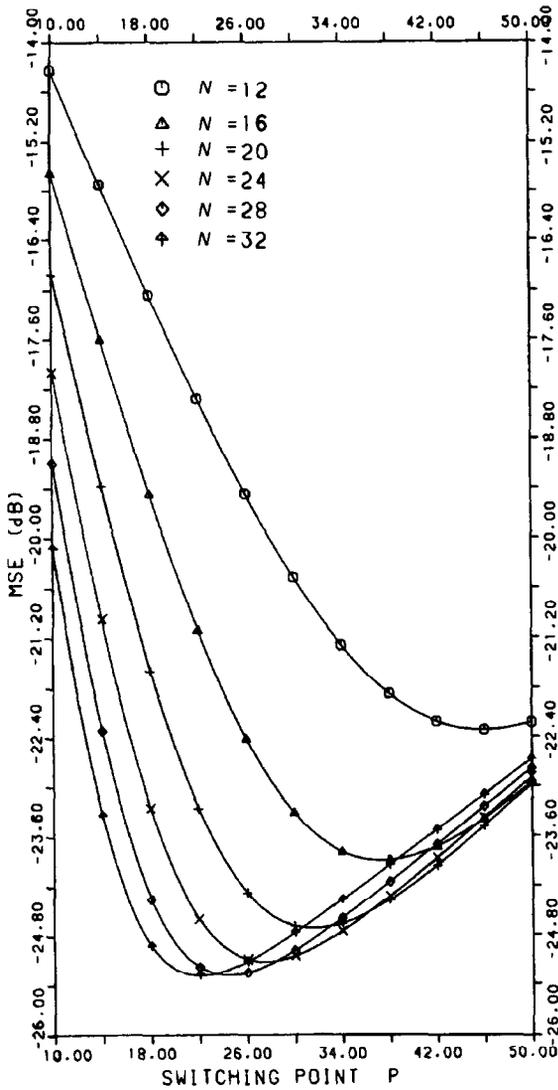


Fig. 6. The MSE versus switching point for varied weight length with SNR = 60 dB, SSR = 20 dB, $\mu_1 = 0.0025$, $\mu_2 = 0.01$.

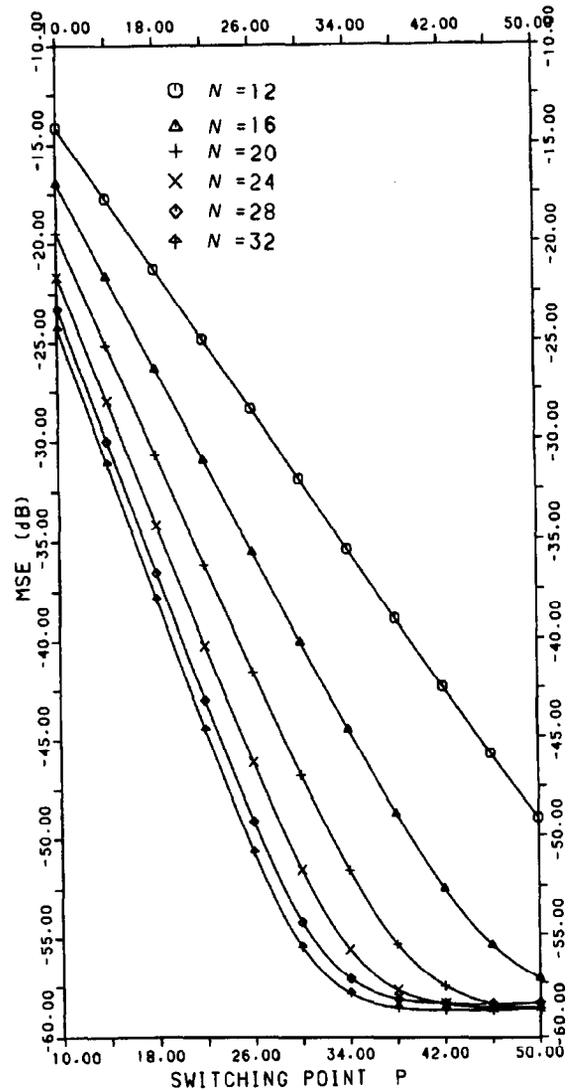


Fig. 7. The MSE versus switching point for varied weight length with SNR = 60 dB, SSR = 0 dB, $\mu_1 = 0.005$, $\mu_2 = 0.005$.

simulation:

$$d(k) = A_1 \exp(j2\pi f_1 k) + A_2 \exp(j2\pi f_2 k) + n(k) \tag{24}$$

and for simplicity T_s is set to unity. Moreover, in the simulation data results, the two frequencies f_1 and f_2 are chosen to be 0.2 and 0.25 (normalized frequency), respectively. Thus from [21], the delay

Δ is chosen to be

$$\Delta = \frac{1}{2|f_1 - f_2|} - \frac{N - 1}{2} \cong 2$$

for $N = 16$. The theoretical and simulation results with parameters, $N = 16$, $L = 2$, $\mu_1 = 0.005$, $p = 30$ and $SSR = 20$ dB are shown in Figs. 9 and 10, for SNR = 20 and 40 dB, respectively. As can be seen in Figs. 9 and 10, the theoretical results agree quite

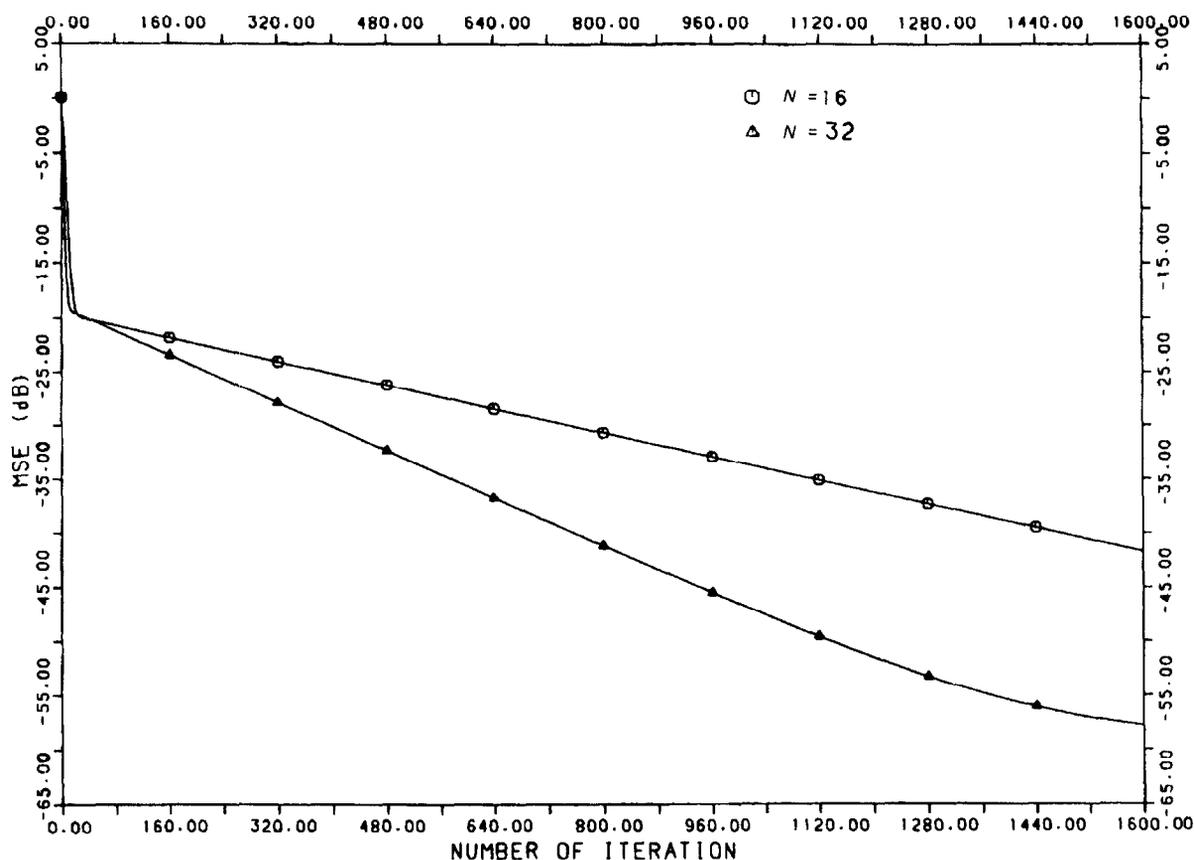


Fig. 8. The convergence rate of the conventional LMS algorithm for tap-weight length being 16 and 32 with SNR = 60 dB, SSR = 20 dB, $\mu_1 = 0.005$.

well with the simulation results. It is noted that in implementing the inverse of \mathbf{R} , in the HLMS algorithm for the simulation data, a recursive formulation shown in [10] is employed. The learning curve is the average of 100 runs for the simulation results.

To see the advantage of the HLMS algorithm, the comparison with the LMS algorithm and the NLMS algorithm are made. These are shown in Figs. 11–13, for varied SSR being 0, 10 and 20 dB, and with parameters $N = 16$, SNR = 40, $p = 30$ and step-size $\mu_2 = 0.01$ in the NLMS algorithm and $\mu_1 = 0.005$, $\mu_2 = 0.01$ in HLMS algorithm. In all of the cases shown in Figs. 11–13, we see that the HLMS algorithm converges to the same steady state MSE as the NLMS algorithm, but has much faster convergence rate in the transient state. The reason of considering relatively large SNR in our

discussion is because in such cases the disparity of eigenvalues of \mathbf{R} will be relatively large and under this circumstance the conventional LMS adaptive algorithm may not perform well. Indeed, this situation is often encountered in the adaptive array beamforming system for jammer suppression, in which we have a broad-band signal buried in broad-band noise with multiple jammers in the underwater environment.

Similar results are observed from the same plots for the HLMS algorithm compared to the LMS algorithm. Since as shown in [1], in the case that SSR is 0 dB (sinusoids with equal power), the LMS algorithm can converge much faster. However, even in this case, the performance of the HLMS algorithm is still compatible with the LMS algorithm. Therefore, we can conclude that the HLMS

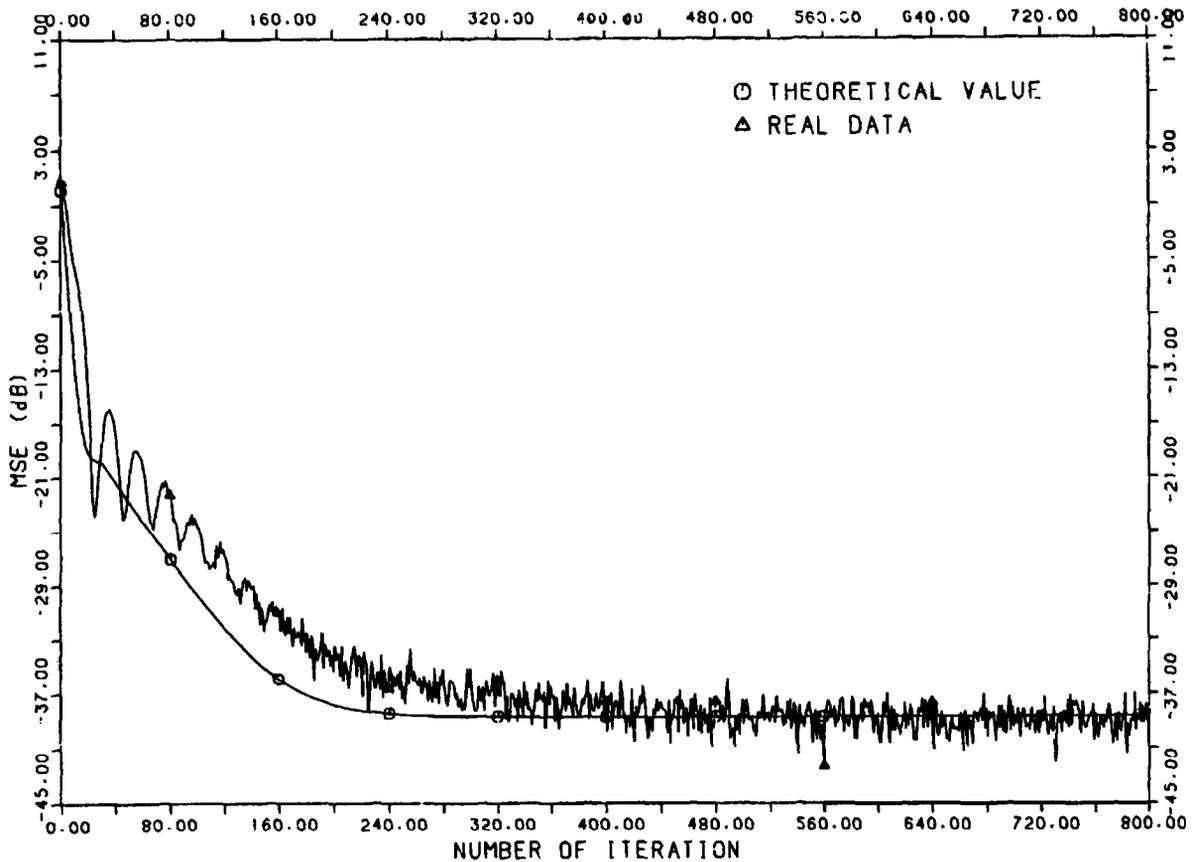


Fig. 9. The comparison of the theoretical and simulation results of the HLMS algorithm with SNR = 40 dB, SSR = 20 dB, $\mu_1 = 0.005$, $\mu_2 = 0.01$ and switching point being 30.

algorithm has better performance and is more robust than the conventional LMS algorithm as well as the NLMS algorithm.

Next, let us discuss the effect when the optimum switching point is not adopted. As shown in Fig. 3, the optimum value of p is 24. Now, if all other parameters are kept the same as those in Fig. 3, but the values of p are chosen to be 14, 24 and 40, the results of the ALE are shown in Fig. 14. From Fig. 14, we observed that the value of the switching point does not affect the convergence property significantly. Therefore, based on this observation, we suggest that the value of p can be chosen proportional to the value of N , filter length. Based on the theoretical and simulation results, we will suggest that the value of switching point in the range of

N to $1.5N$ will have a better performance. This is reasonable, because with the data number equal to the tap-weight length or slightly greater than it, the input data will be sufficient to estimate the power or matrix \mathbf{R} .

5. Conclusion

In this paper, the performance of the HLMS algorithm has been investigated. For evaluating the performance of the HLMS algorithm, an analytical expression, in terms of transient MSE, with application to ALE, was derived. Moreover, the results of simulation data verified the accuracy of the theoretical analysis. In general, the HLMS

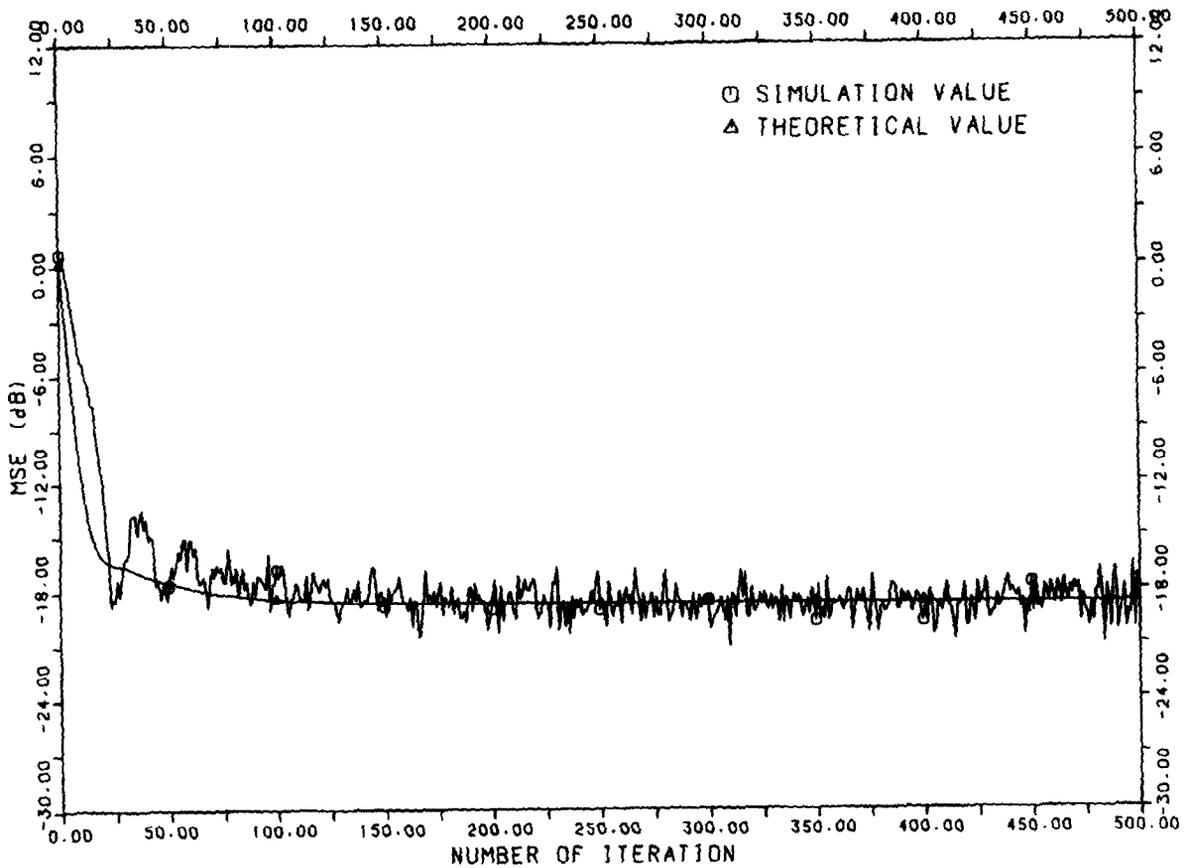


Fig. 10. The comparison of the theoretical and simulation results of the HLMS algorithm with SNR = 20 dB, SSR = 20 dB, $\mu_1 = 0.005$, $\mu_2 = 0.01$ and switching point being 30.

algorithm can yield more robust performance than the conventional LMS algorithm and the NLMS algorithm.

The selection of switching point is also addressed using the numerical approach. Many interesting characteristics were observed, which showed the relationship to the relevant parameters of the ALE. However, an explicit expression of the optimum switching point is still a challenge in our future study. Fortunately, the value of the switching point is not vital as to affect the convergence property significantly in our study.

Appendix A

By proceeding in a similar manner as in [7], Eq. (11b) can also be derived. Since the input autocor-

relation matrix \mathbf{R} can be represented by

$$\mathbf{R} = \mathbf{Q}\mathbf{A}\mathbf{Q}^H = \sum_{i=1}^N \lambda_i \mathbf{q}_i \mathbf{q}_i^H, \quad (\text{A.1})$$

where λ_i are the eigenvalues of \mathbf{R} and \mathbf{q}_i are the corresponding eigenvectors defined in Eqs. (6). Thus, the matrix $(\mathbf{I} - 2\mu_1 \mathbf{R})^k$ can be expressed as

$$(\mathbf{I} - 2\mu_1 \mathbf{R})^k = \sum_{i=0}^N (1 - 2\mu_1 \lambda_i)^k \mathbf{q}_i \mathbf{q}_i^H. \quad (\text{A.2})$$

For consistency, we let $M_w(0)$ be a null vector which we used to derive Eq. (11a). Using this fact, Eq. (9b) can be simplified as

$$M_w(k) = (1 - 2\mu_2)^{k-p} 2\mu_1 \sum_{i=0}^{p-1} (\mathbf{I} - 2\mu_1 \mathbf{R})^i \mathbf{p} + 2\mu_2 \mathbf{R}^{-1} \sum_{i=p}^{k-1} (1 - 2\mu_2)^{k-i-1} \mathbf{p}, \quad (\text{A.3})$$

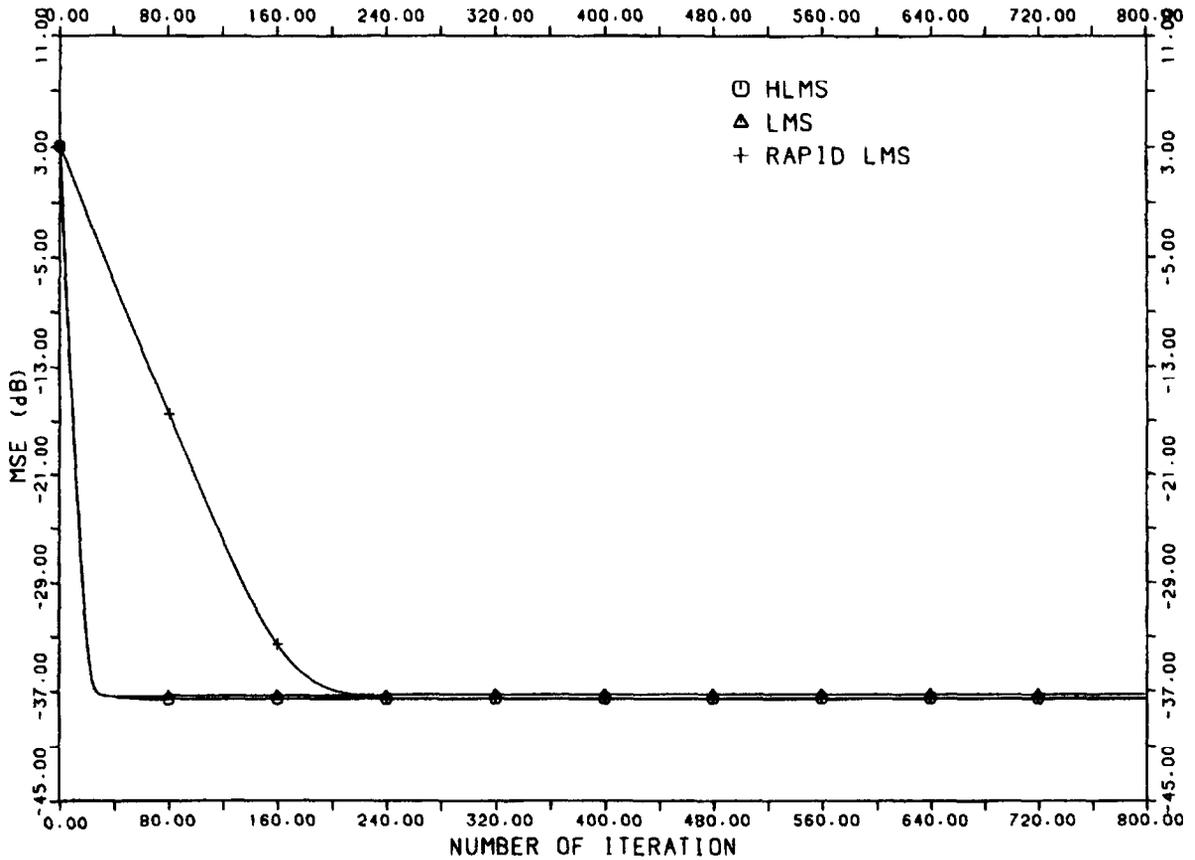


Fig. 11. The performance of the HLMS, conventional LMS and NLMS algorithms with SNR = 40 dB, SSR = 0 dB, $\mu_1 = 0.005$, $\mu_2 = 0.005$, $N = 16$ and switching point = 30.

where the cross-correlation vector p was defined in Eq. (7). Applying Eq. (A.2) and Eq. (7) to Eq. (A.3), we have

$$\begin{aligned}
 M_w(k) &= (1 - 2\mu_2)^{k-p} \sqrt{N} \sum_{i=1}^L \beta_i \frac{\sigma_{s_i}^2}{\lambda_i} \{1 - (1 - 2\mu_1 \lambda_i)^p\} \mathbf{q}_i \\
 &= \sum_{i=1}^L \beta_i \frac{\sigma_{s_i}^2}{\lambda_i} \{1 - (1 - 2\mu_1 \lambda_i)^p (1 - 2\mu_2)^{k-p}\} \mathbf{v}_i.
 \end{aligned} \tag{A.4}$$

Hence, the proof of Eq. (11b) is also completed.

Appendix B

The derivation of the weight covariance matrix recursive equation of the equivalent time-domain expression of the complex NLMS algorithm can be obtained using the Gaussian Moment Factoring Theorem. This theorem states [16] that for given complex Gaussian random variables, x_1, x_2, x_3 and x_4 , the following relationship holds:

$$\begin{aligned}
 E[x_1 x_2^* x_3 x_4^*] &= E[x_1 x_2^*] E[x_3 x_4^*] \\
 &\quad + E[x_1 x_4^*] E[x_2^* x_3].
 \end{aligned} \tag{B.1}$$

Now, we consider the weight covariance matrix,

$$\text{Cov}[\mathbf{w}(k+1)] = E\{[\mathbf{w}(k+1) - M_w(k+1)]$$

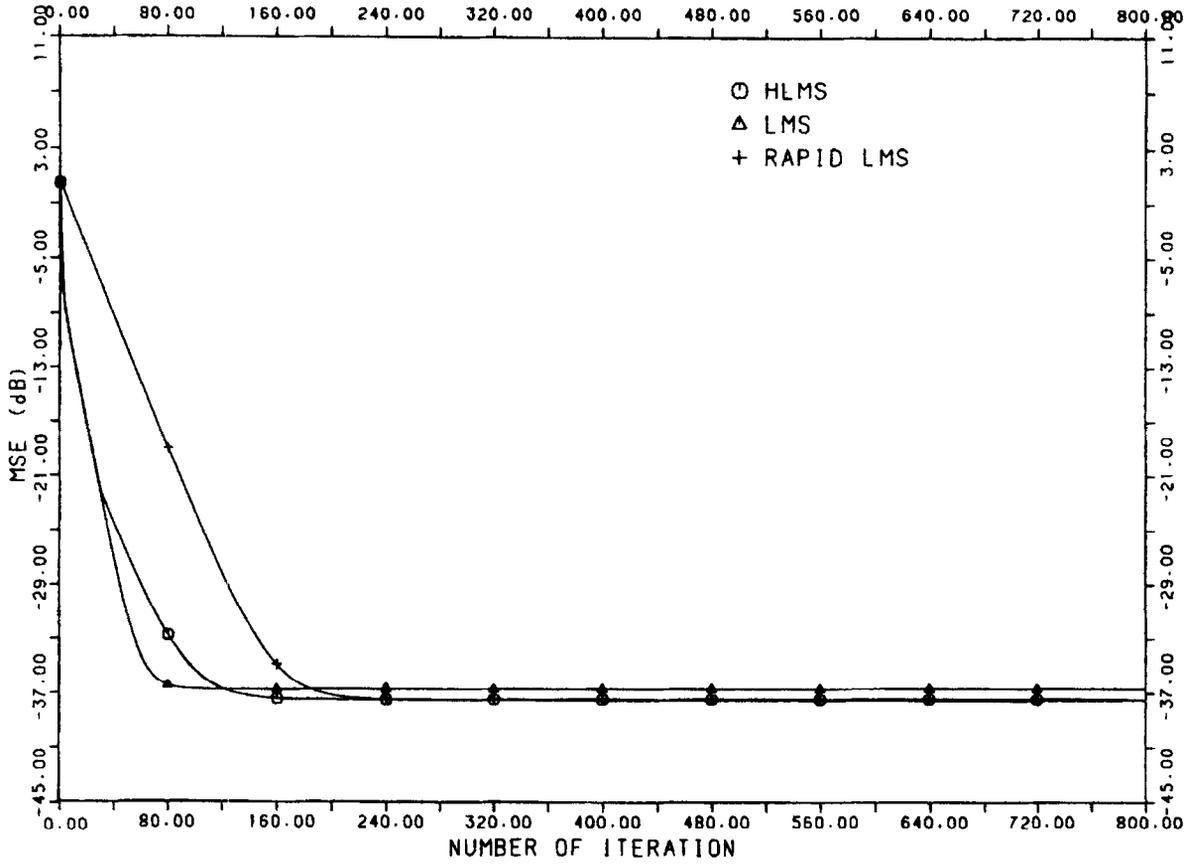


Fig. 12. The performance comparison of the HLMS, conventional LMS and NLMS algorithms with SNR = 40 dB, SSR = 10 dB, $\mu_1 = 0.005$, $\mu_2 = 0.01$, $N = 16$ and switching point = 30.

$$\times [\mathbf{w}(k+1) - M_{\mathbf{w}(k+1)}]^\text{H}. \quad (\text{B.2})$$

Assuming that $\mathbf{w}(k)$ depends only on $\{d(n), \mathbf{x}(n); n = 0, 1, \dots, k-1\}$. Moreover, the different index elements of the sequence of data vectors and also desired signal samples are assumed to be statistically independent. Thus, from Eq. (2b), the mean weight vector recursive equation can be written as

$$\begin{aligned} M_{\mathbf{w}(k+1)} &= \{I - 2\mu_2 \mathbf{R}^{-1} E[\mathbf{x}^*(k)\mathbf{x}^\text{T}(k)]\} E[\mathbf{w}(k)] \\ &\quad + 2\mu_2 \mathbf{R}^{-1} \mathbf{p} \\ &= (1 - 2\mu_2) M_{\mathbf{w}(k)} + 2\mu_2 \mathbf{R}^{-1} \mathbf{p}. \end{aligned} \quad (\text{B.3})$$

Substituting Eqs. (2b) and (B.3) into Eq. (B.2) and

after some manipulation, we obtain

$$\begin{aligned} &\text{Cov}[\mathbf{w}(k+1)] \\ &= \text{Cov}[\mathbf{w}(k)] - 4\mu_2 \text{Cov}[\mathbf{w}(k)] \\ &\quad + 4\mu_2^2 \mathbf{R}^{-1} E\{\mathbf{x}^*(k)\mathbf{x}^\text{T}(k) E[\mathbf{w}(k)\mathbf{w}^\text{H}(k)] \\ &\quad \quad \times \mathbf{x}^*(k)\mathbf{x}^\text{T}(k)\} \mathbf{R}^{-1} \\ &\quad - 4\mu_2^2 \mathbf{R}^{-1} E\{\mathbf{x}^*(k)\mathbf{x}^\text{T}(k) M_{\mathbf{w}(k)} d^*(k) \mathbf{x}^\text{T}(k)\} \mathbf{R}^{-1} \\ &\quad + 4\mu_2^2 M_{\mathbf{w}(k)} \mathbf{p}^\text{H} \mathbf{R}^{-1} - 4\mu_2^2 M_{\mathbf{w}(k)} M_{\mathbf{w}(k)}^\text{H} \\ &\quad - 4\mu_2^2 \mathbf{R}^{-1} E\{d(k)\mathbf{x}^*(k) M_{\mathbf{w}(k)}^\text{H} \mathbf{x}^*(k)\mathbf{x}^\text{T}(k)\} \mathbf{R}^{-1} \\ &\quad + 4\mu_2^2 \mathbf{R}^{-1} \mathbf{p}^\text{H} M_{\mathbf{w}(k)} \\ &\quad + 4\mu_2^2 \mathbf{R}^{-1} E\{d(k)\mathbf{x}^*(k) d^*(k) \mathbf{x}^\text{T}(k)\} \mathbf{R}^{-1} \\ &\quad - 4\mu_2^2 \mathbf{R}^{-1} \mathbf{p} \mathbf{p}^\text{H} \mathbf{R}^{-1}. \end{aligned} \quad (\text{B.4})$$

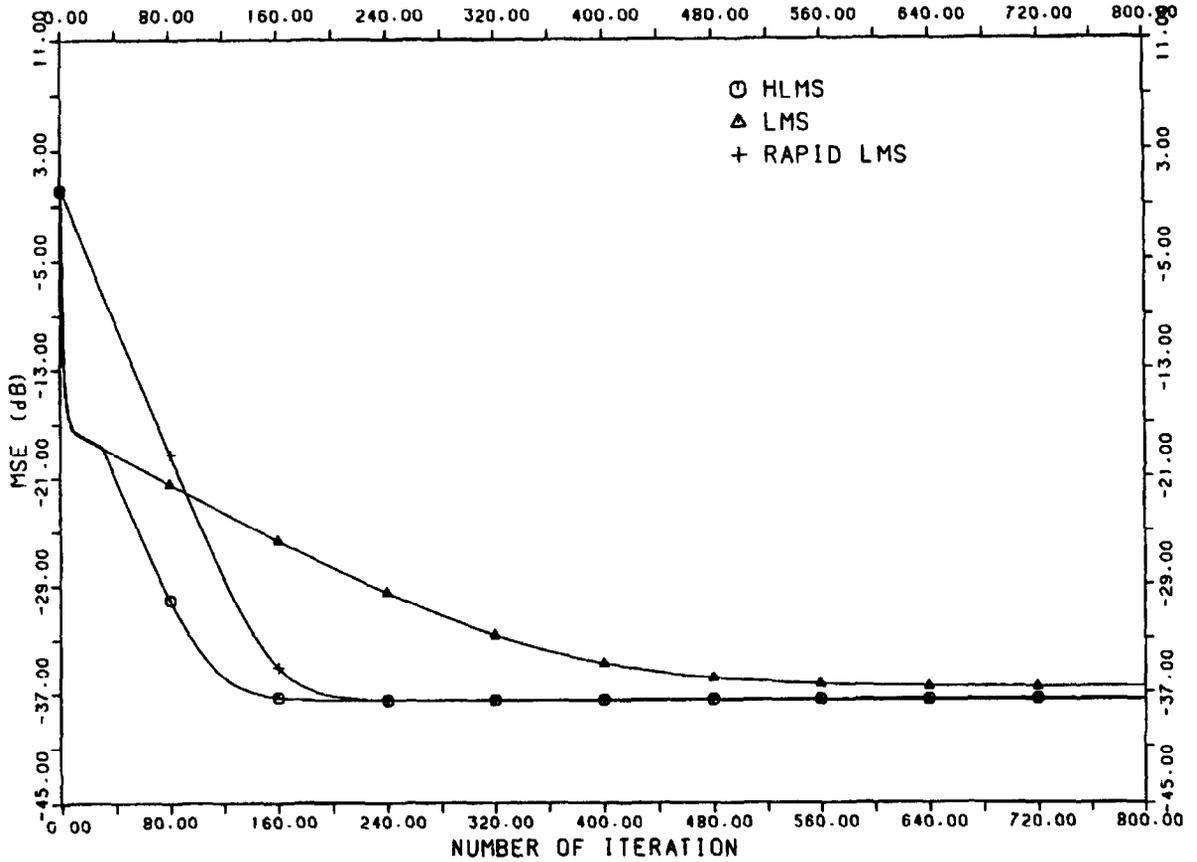


Fig. 13. The performance comparison of the HLMS, conventional LMS and NLMS algorithms with SNR = 40 dB, SSR = 20 dB, $\mu_1 = 0.005$, $\mu_2 = 0.01$, $N = 16$ and switching point = 30.

Further evaluation of the third, fourth, seventh and ninth terms on the right-hand side of Eq. (B.4) require the moment theorem which was described in Eq. (B.1).

Now, applying Eq. (B.1) to the third term on the right-hand side of Eq. (B.4) we have the similar results as in [7]:

$$\begin{aligned}
 & E\{\mathbf{x}^*(k)\mathbf{x}^T(k)E[\mathbf{w}(k)\mathbf{w}^H(k)]\mathbf{x}^*(k)\mathbf{x}^T(k)\} \\
 &= \mathbf{R}E[\mathbf{w}(k)\mathbf{w}^H(k)]\mathbf{R} + \mathbf{R}\text{Tr}\{\mathbf{R}E[\mathbf{w}(k)\mathbf{w}^H(k)]\}.
 \end{aligned} \tag{B.5}$$

Similarly, applying Eq. (B.1) to the fourth term on the right-hand side of Eq. (B.4), we have

$$\begin{aligned}
 & E\{\mathbf{x}^*(k)\mathbf{x}^T(k)M_{\mathbf{w}(k)}d^*(k)\mathbf{x}^T(k)\} \\
 &= \mathbf{R}M_{\mathbf{w}(k)}^H\mathbf{p} + \mathbf{p}^H M_{\mathbf{w}(k)}\mathbf{R}.
 \end{aligned} \tag{B.6}$$

Again, since the seventh term of right-hand side of (B.4) is the transpose matrix of the fourth term, thus,

$$\begin{aligned}
 & E\{\mathbf{x}^*(k)d(k)M_{\mathbf{w}(k)}^H\mathbf{x}^*(k)\mathbf{x}^T(k)\} \\
 &= \mathbf{p}M_{\mathbf{w}(k)}^H\mathbf{R} + \mathbf{R}M_{\mathbf{w}(k)}^H\mathbf{p}.
 \end{aligned} \tag{B.7}$$

Finally, from (B.4), the ninth term on the right-hand side of (B.4) will be

$$E\{d(k)\mathbf{x}^*(k)\mathbf{x}^T(k)d^*(k)\} = \mathbf{p}\mathbf{p}^H + \sigma_d^2\mathbf{R}. \tag{B.8}$$

Now, applying the results from Eqs. (B.5)–(B.8) to Eq. (B.4) yields the weight covariance matrix

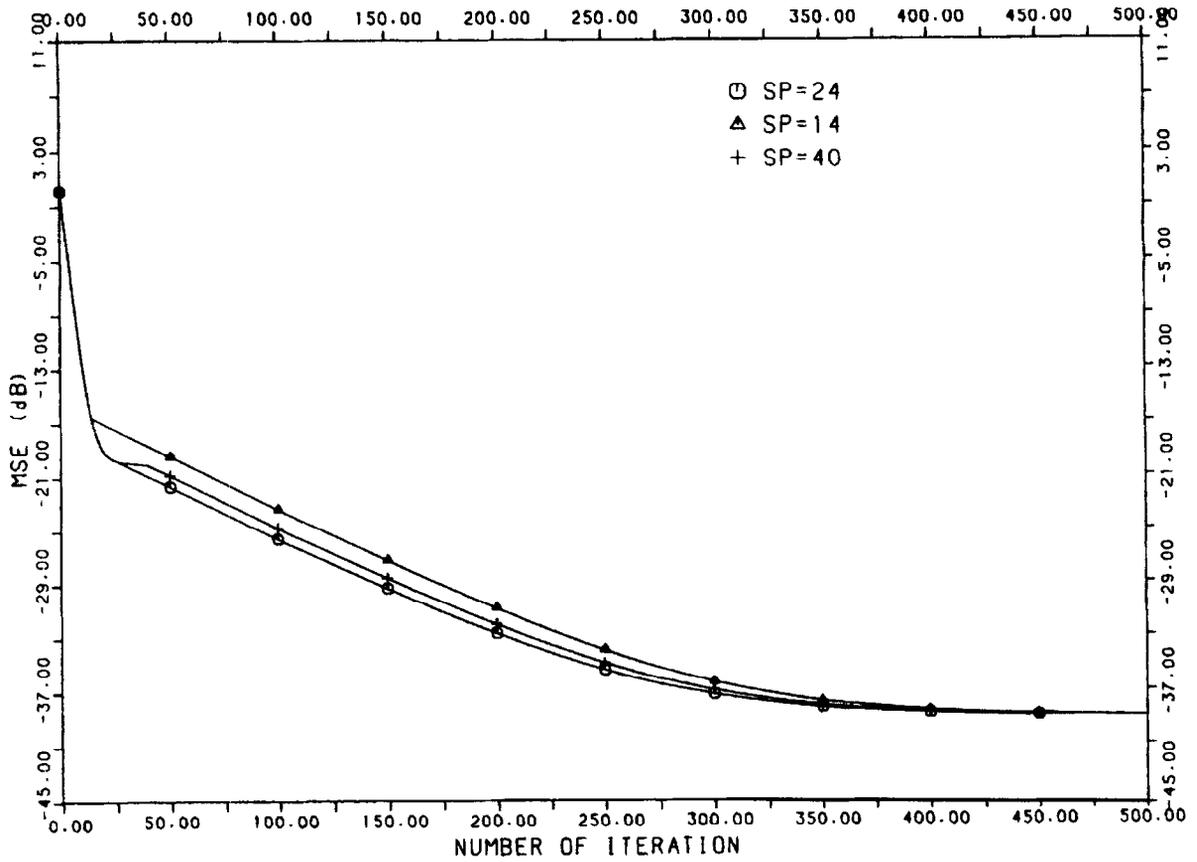


Fig. 14. The effect of the improper selection of the switching point with SNR = 40 dB, SSR = 20 dB, $N = 16$, $\mu_1 = 0.005$, $\mu_2 = 0.005$.

recursive equation, i.e.,

$$\begin{aligned}
 \text{Cov}\{\mathbf{w}(k+1)\} &= (1 - 4\mu_2 + 4\mu_2^2)\text{Cov}\{\mathbf{w}(k)\} \\
 &+ 4\mu_2^2 \text{Tr}\{\mathbf{R}E[\mathbf{w}(k)\mathbf{w}^H(k)]\}\mathbf{R}^{-1} \\
 &+ 4\mu_2^2 \mathbf{R}^{-1}\{\sigma_d^2 - \mathbf{M}_{\mathbf{w}(k)}^H \mathbf{p} - \mathbf{p}^H \mathbf{M}_{\mathbf{w}(k)}\}. \quad (\text{B.9})
 \end{aligned}$$

Using the following relationship:

$$\begin{aligned}
 \text{Tr}\{\mathbf{R}E[\mathbf{w}(k)\mathbf{w}^H(k)]\} &= \text{Tr}\{\mathbf{R}[\text{Cov}\{\mathbf{w}(k)\} + \mathbf{M}_{\mathbf{w}(k)} \mathbf{M}_{\mathbf{w}(k)}^H]\}, \quad (\text{B.10})
 \end{aligned}$$

Eq. (B.9) can be rewritten as

$$\begin{aligned}
 \text{Cov}\{\mathbf{w}(k+1)\} &= (1 - 2\mu_2)^2 \text{Cov}\{\mathbf{w}(k)\} \\
 &+ 4\mu_2^2 \text{Tr}\{\mathbf{R} \text{Cov}\{\mathbf{w}(k)\}\}\mathbf{R}^{-1} \\
 &+ 4\mu_2^2 \mathbf{R}^{-1}\{\sigma_d^2 - \mathbf{M}_{\mathbf{w}(k)}^H \mathbf{p} \\
 &- \mathbf{p}^H \mathbf{M}_{\mathbf{w}(k)} + \mathbf{M}_{\mathbf{w}(k)}^H \mathbf{R} \mathbf{M}_{\mathbf{w}(k)}\}. \quad (\text{B.11})
 \end{aligned}$$

Appendix C

In this appendix, the values of $\gamma_i(p)$, in terms of its initial value, derived in [1] is summarized in what follows. For convenience, we rewrite Eq. (19a) (for

$L = 2$) in a matrix form (with reduced rank), i.e.

$$\begin{aligned}\gamma(p) &= [\gamma_1(p) \ \gamma_2(p) \ \gamma_3(p)]^T, \\ &= [\mathbf{I} - \mathbf{H}]\gamma(p-1) + 4\mu_1^2 \eta_0(p-1)\lambda,\end{aligned}\quad (\text{C.1})$$

where

$$\mathbf{H} = \begin{bmatrix} 4\mu_1 \lambda_1 (1 - 2\mu_1 \lambda_1) & -4\mu_1^2 \lambda_1 \lambda_2 & -4\mu_1^2 (N-2) \lambda_1 \lambda_3 \\ -4\mu_1^2 \lambda_1 \lambda_2 & 4\mu_1 \lambda_2 (1 - 2\mu_1 \lambda_2) & -4\mu_1^2 (N-2) \lambda_2 \lambda_3 \\ -4\mu_1^2 \lambda_1 \lambda_3 & -4\mu_1^2 \lambda_3 \lambda_2 & 4\mu_1 \lambda_3 [1 - \mu_1 (N-1) \lambda_3] \end{bmatrix}, \quad (\text{C.2})$$

$\lambda = [\lambda_1 \ \lambda_2 \ \lambda_3]^T$ and $\eta_0(p-1)$ was defined in Eq. (21a) with $k = p-1$. Note that \mathbf{H} is not a symmetric matrix; however, we can find a similarity transformation for \mathbf{H} :

$$\mathbf{\Pi} = \text{diag}(1 \ 1 \ \sqrt{N-2}), \quad (\text{C.3})$$

such that \mathbf{H} can be transformed to be as

$$\mathbf{A} = \mathbf{\Pi} \mathbf{H} \mathbf{\Pi}^{-1} = \begin{bmatrix} 4\mu_1 \lambda_1 (1 - 2\mu_1 \lambda_1) & -4\mu_1^2 \lambda_1 \lambda_2 & -4\mu_1^2 \sqrt{N-2} \lambda_1 \lambda_3 \\ -4\mu_1^2 \lambda_1 \lambda_2 & 4\mu_1 \lambda_2 (1 - 2\mu_1 \lambda_2) & -4\mu_1^2 \sqrt{N-2} \lambda_2 \lambda_3 \\ -4\mu_1^2 \sqrt{N-2} \lambda_1 \lambda_3 & -4\mu_1^2 \sqrt{N-2} \lambda_2 \lambda_3 & 4\mu_1 \lambda_3 [1 - \mu_1 (N-1) \lambda_3] \end{bmatrix}. \quad (\text{C.4})$$

Furthermore, for convenience, let us define a new vector, $\mathbf{c}(p)$,

$$\mathbf{c}(p) = \mathbf{\Pi} \gamma(p) = (\mathbf{I} - \mathbf{A}) \mathbf{c}(p-1) + 4\mu_1^2 \eta_0(p-1) \mathbf{b}, \quad (\text{C.5})$$

where

$$\mathbf{b} = \mathbf{\Pi} \lambda = [\lambda_1 \ \lambda_2 \ \sqrt{N-2} \lambda_3]^T. \quad (\text{C.6})$$

Now, since \mathbf{A} is symmetric, it can be factorized as

$$\mathbf{A} = \mathbf{S} \mathbf{\Sigma} \mathbf{S}^T, \quad (\text{C.7})$$

where \mathbf{S} is a unitary matrix and $\mathbf{\Sigma}$ is a diagonal matrix which can be defined as

$$\mathbf{S} = (s_1 \ s_2 \ s_3)$$

and

$$\mathbf{\Sigma} = \text{diag}(\alpha_1 \ \alpha_2 \ \alpha_3), \quad (\text{C.8})$$

respectively, with $\{\alpha_i; i = 1, 2, 3\}$ being the eigenvalues of \mathbf{A} and s_i is the associated eigenvector of α_i . Now, starting with $\mathbf{c}(0)$, the initial vector of $\mathbf{c}(p)$ and iteratively substituting in Eq. (C.5), we have

$$\begin{aligned}\mathbf{c}(p) &= (\mathbf{I} - \mathbf{A})^p \mathbf{c}(0) + 4\mu_1^2 \xi_{\min} \sum_{i=0}^{p-1} (\mathbf{I} - \mathbf{A})^i \mathbf{b} + 4\mu_1^2 \sum_{i=0}^{p-1} \frac{N(\sigma_{s_1}^2)^2}{\lambda_1} (\mathbf{I} - \mathbf{A})^{p-i-1} (1 - 2\mu_1 \lambda_1)^{2i} \mathbf{b} \\ &\quad + 4\mu_1^2 \sum_{i=0}^{p-1} \frac{N(\sigma_{s_2}^2)^2}{\lambda_2} (\mathbf{I} - \mathbf{A})^{p-i-1} (1 - 2\mu_1 \lambda_2)^{2i} \mathbf{b}.\end{aligned}\quad (\text{C.9})$$

It is noted that in obtaining Eq. (C.9), Eqs. (20a) and (21a) have been used. Now, since \mathbf{S} is a unitary

matrix, we have

$$(\mathbf{I} - \mathbf{A})^p = \mathbf{S} \begin{bmatrix} (1 - \alpha_1)^p & 0 & 0 \\ 0 & (1 - \alpha_2)^p & 0 \\ 0 & 0 & (1 - \alpha_3)^p \end{bmatrix} \mathbf{S}^T. \quad (\text{C.10})$$

Apply Eq. (C.10) to Eq. (C.9), we have

$$\begin{aligned} \mathbf{c}(p) = & \mathbf{S} \begin{bmatrix} (1 - \alpha_1)^p & 0 & 0 \\ 0 & (1 - \alpha_2)^p & 0 \\ 0 & 0 & (1 - \alpha_3)^p \end{bmatrix} \mathbf{S}^T \mathbf{c}(0) \\ & + 4\mu_1^2 \xi_{\min} \mathbf{S} \begin{bmatrix} \frac{1 - (1 - \alpha_1)^p}{\alpha_1} & 0 & 0 \\ 0 & \frac{1 - (1 - \alpha_2)^p}{\alpha_2} & 0 \\ 0 & 0 & \frac{1 - (1 - \alpha_3)^p}{\alpha_3} \end{bmatrix} \mathbf{S}^T \mathbf{b} \\ & + 4\mu_1^2 N \frac{(\sigma_{s_1}^2)^2}{\lambda_1} \mathbf{S} \begin{bmatrix} \frac{(1 - 2\mu_1 \lambda_1)^{2p} - (1 - \alpha_1)^p}{(1 - 2\mu_1 \lambda_1)^2 - (1 - \alpha_1)} & 0 & 0 \\ 0 & \frac{(1 - 2\mu_1 \lambda_1)^{2p} - (1 - \alpha_2)^p}{(1 - 2\mu_1 \lambda_1)^2 - (1 - \alpha_2)} & 0 \\ 0 & 0 & \frac{(1 - 2\mu_1 \lambda_1)^{2p} - (1 - \alpha_3)^p}{(1 - 2\mu_1 \lambda_1)^2 - (1 - \alpha_3)} \end{bmatrix} \mathbf{S}^T \mathbf{b} \\ & + 4\mu_1^2 N \frac{(\sigma_{s_2}^2)^2}{\lambda_2} \mathbf{S} \begin{bmatrix} \frac{(1 - 2\mu_1 \lambda_2)^{2p} - (1 - \alpha_1)^p}{(1 - 2\mu_1 \lambda_2)^2 - (1 - \alpha_1)} & 0 & 0 \\ 0 & \frac{(1 - 2\mu_1 \lambda_2)^{2p} - (1 - \alpha_2)^p}{(1 - 2\mu_1 \lambda_2)^2 - (1 - \alpha_2)} & 0 \\ 0 & 0 & \frac{(1 - 2\mu_1 \lambda_2)^{2p} - (1 - \alpha_3)^p}{(1 - 2\mu_1 \lambda_2)^2 - (1 - \alpha_3)} \end{bmatrix} \mathbf{S}^T \mathbf{b}. \end{aligned} \quad (\text{C.11})$$

Finally, the elements of $\gamma(p)$ can be obtained from Eq. (C.12):

$$\gamma(p) = \mathbf{\Pi}^{-1} \mathbf{c}(p). \quad (\text{C.12})$$

Acknowledgements

The financial support of partial work of this paper by the National Science Council, Republic of China, through the research project NSC79-0404-E-110-11, is greatly acknowledged.

References

- [1] S.J. Chern, "The power effect on the convergence rate of the complex LMS adaptive line enhancer", *Proc. National Science Council, R.O.C., Part A: Phys. Engrg.*, Vol. 13, No. 1, 1989, pp. 32–39.

- [2] S.J. Chern and B.Y. Chen, "The study of hybrid adaptive least mean square (LMS) adaptive filtering algorithm with variable step-size", *Proc. National Science Council, R.O.C., Part A: Phys. Engrg.*, Vol. 13, No. 3, 1989, pp. 130–144.
- [3] S.J. Chern and S.C. Chen, "High resolution frequency estimation using the sub-optimal eigen-decomposition method and adaptive filtering", *Proc. National Science Council, R.O.C., Part A: Phys. Engrg.*, Vol. 14, No. 4, 1990, pp. 304–316.
- [4] S.J. Chern and K.M. Wong, "Transformed domain adaptive filtering with application to adaptive line enhancer", *Proc. 1987 Telecom. Symposium, Taiwan, R.O.C.*, pp. 131–140.
- [5] N.I. Cho, C.H. Choi and S.U. Lee, "Adaptive line enhancement by using an IIR notch filter", *IEEE Trans. Acoust. Speech Signal Process.*, Vol. ASSP-37, No. 4, 1989, pp. 585–589.
- [6] A. Feuer and E. Weinstein, "Convergence analysis of LMS filters with uncorrelated Gaussian data", *IEEE Trans. Acoust. Speech Signal Process.*, Vol. ASSP-33, No. 1, 1985, pp. 222–228.
- [7] B. Fisher and N.J. Bershad, "The complex LMS algorithm – Transient weight mean and covariance with applications to the ALE", *IEEE Trans. Acoust. Speech Signal Process.*, Vol. ASSP-31, No. 1, 1983, pp. 34–44.
- [8] L.J. Griffiths, "Rapid measurement of digital instantaneous frequency", *IEEE Trans. Acoust. Speech Signal Process.*, Vol. ASSP-23, No. 2, 1975, pp. 207–222.
- [9] R.W. Harris, D.M. Chabries and F.A. Bishop, "A variable step-size (VS) adaptive filter algorithm", *IEEE Trans. Acoust. Speech Signal Process.*, Vol. ASSP-34, No. 2, 1986, pp. 309–316.
- [10] S. Haykin, *Adaptive Filter Theory*, Prentice-Hall, Englewood Cliffs, NJ, 1986.
- [11] C.P. Kwong, "Dual sign algorithm for adaptive filtering", *IEEE Trans. Commun.*, Vol. COM-34, No. 12, 1986, pp. 1272–1275.
- [12] J.C. Lee and C.H. Un, "Performance of transform-domain LMS adaptive digital filters", *IEEE Trans. Acoust. Speech Signal Process.*, Vol. ASSP-34, No. 3, 1986, pp. 499–510.
- [13] S.S. Narayan, A.M. Peterson and M.J. Narasimha, "Transform domain LMS algorithm", *IEEE Trans. Acoust. Speech Signal Process.*, Vol. ASSP-31, No. 3, 1983, pp. 609–615.
- [14] B. Proat and T. Kailath, "Normalized lattice algorithm for least-squares FIR system identification", *IEEE Trans. Acoust. Speech Signal Process.*, Vol. ASSP-31, No. 1, 1983, pp. 122–128.
- [15] S.S. Reddi, "A time-domain adaptive algorithm for rapid convergence", *Proc. IEEE*, Vol. 72, No. 4, 1984, pp. 533–535.
- [16] I.S. Reed, "On a moment theory for complex Gaussian processes", *IRE Trans. Inform. Theory*, 1962, pp. 194–196.
- [17] J.R. Treichler, "Transient and convergent behaviour of the adaptive line enhancer", *IEEE Trans. Acoust. Speech Signal Process.*, Vol. ASSP-27, No. 1, 1979, pp. 53–62.
- [18] B. Widrow, "Adaptive Filters", in: R.E. Kalman and N.D. Claris, eds, *Aspects of Network and System Theory*, Holt, Rinehart and Winston, New York, 1971, pp. 563–587.
- [19] B. Widrow, J.R. Glover, J.M. McCool, J. Kaunitz, C.S. Willians, R.H. Hearn, J.R. Zeidler, E. Dong and R.C. Goodline, "Adaptive noise cancelling: Principles and applications", *Proc. IEEE*, Vol. 63, No. 12, 1975, pp. 1692–1716.
- [20] B. Widrow and S.D. Stearn, *Adaptive Signal Processing*, Prentice-Hall, Englewood Cliffs, NJ, 1985.
- [21] J.R. Zeidler, E.H. Satorins, D.M. Charbies and H.T. Wexler, "Adaptive enhancement of multiple sinusoids in uncorrelated noise", *IEEE Trans. Acoust. Speech Signal Process.*, Vol. ASSP-26, 1978, pp. 240–254.