

# Selective-Tap Adaptive Filtering With Performance Analysis for Identification of Time-Varying Systems

Andy W. H. Khong, *Member, IEEE*, and Patrick A. Naylor, *Senior Member, IEEE*

**Abstract**—Selective-tap algorithms employing the MMax tap selection criterion were originally proposed for low-complexity adaptive filtering. The concept has recently been extended to multichannel adaptive filtering and applied to stereophonic acoustic echo cancellation. This paper first briefly reviews least mean square versions of MMax selective-tap adaptive filtering and then introduces new recursive least squares and affine projection MMax algorithms. We subsequently formulate an analysis of the MMax algorithms for time-varying system identification by modeling the unknown system using a modified Markov process. Analytical results are derived for the tracking performance of MMax selective tap algorithms for normalized least mean square, recursive least squares, and affine projection algorithms. Simulation results are shown to verify the analysis.

**Index Terms**—Acoustic echo cancellation, misalignment analysis, partial-updating algorithms, time-varying system identification.

## I. INTRODUCTION

ADAPTIVE filters with finite-impulse response (FIR) are now widely used in many applications of signal processing in general and telecommunications in particular. The least-mean-square (LMS) algorithm and the normalized version (NLMS) [1] are the most common in practice because of their straightforward implementation and relatively low complexity compared to the better performing but substantially more complex least squares algorithms. The demands made of adaptive filters by the deployment of new technologies call for ever-increasing performance, modeling of longer impulse responses, and lower computational complexity. Examples include acoustic echo cancellation, where support for over 2000 taps can be required, and network gateways, where very high density network echo cancellers are desirable with, say, 100 channels of 128-ms cancellers per processor core.

In the past, this rate of operation was considered high for typical telecommunications end-user equipment, and researchers were therefore motivated to seek techniques that could reduce the computational complexity of adaptation without significantly degrading effectiveness in terms of its convergence rate or steady-state misadjustment. More recently, the computational capability of low-cost processing hardware has increased very rapidly so that a typical NLMS implementation would

not be seen as a heavy computational demand. However, new pressures on product design have emerged: the increase of user mobility imposes a requirement of low power consumption in portable battery-powered equipment; the growth of telecommunications usage imposes a requirement of high-density implementation for infrastructure equipment so that the number of simultaneous echo cancellers of given tap length that can be run within a specified MIP-budget (millions of instructions per second) is maximized. Both these requirements renew the motivation for low computational complexity, even with today's high-speed processors. As a consequence, a significant focus for adaptive filter research in recent years has been to reduce the computational complexity of tap updates per iteration for applications requiring such high density or low cost. A result of this work is a class of partial-update adaptive filtering algorithms that share the characteristic of executing tap update operations on only a subset of the filter coefficients at each iteration. This can be achieved by selecting the taps to update at each iteration by one of several criteria. Examples include decimation in the space of the tap-update vector such as in sequential-LMS [2], temporal decimation such as in periodic-LMS [2], imposition of a sparse approximation to the tap-update vector such as in MMax normalized least-mean-square (MMax-NLMS) [3], [4], and generalized block-based approaches known as selective partial-update algorithms [5] that can be built around both NLMS and the affine projection algorithm [6].

Recently, partial-update algorithms have found applications in both network echo cancellation (NEC) and stereophonic acoustic echo cancellation (SAEC). In NEC, the echo path impulse response is dominated by regions where magnitudes are close to zero making the impulse response sparse. Adaptive algorithms such as the proportionate NLMS (PNLMS) [7] and its improved version (IPNLMS) [8] are employed which exploit the sparse nature of network impulse responses such that each filter coefficient is updated independently using a variable step-size proportional to the estimated filter coefficient. In such cases where the system to be identified is sparse, the use of sparse-tap adaptive algorithms can reduce complexity by modeling only the active regions of the sparse system [9]. A partial update scheme has also been incorporated into PNLMS [10] to achieve fast convergence with low computational complexity for sparse system identification. Although partial-update algorithms were originally proposed to address computational complexity issues as has been discussed, more recently a class of selective-tap algorithms have been applied to stereophonic acoustic echo cancellation (SAEC) [11], [12] giving improved performance. The exclusive-maximum (XM) tap selection criterion addresses the ill-conditioning of the SAEC problem by selecting taps for updating so as to maximize jointly the

Manuscript received November 22, 2005; revised February 17, 2007. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Shoji Makino.

The authors are with the Department of Electrical and Electronic Engineering, Imperial College London, London SW7 2AZ, U.K. (e-mail: andy.khong@imperial.ac.uk; p.naylor@imperial.ac.uk).

Digital Object Identifier 10.1109/TASL.2007.896671

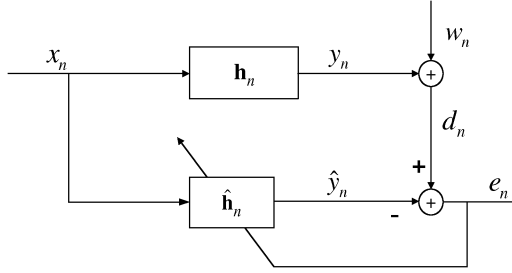


Fig. 1. System identification structure.

$l_1$ -norm of the selected taps over both channels subject to the exclusivity constraint that taps with the same index may not be selected in both channels. In [13] and [14], partial updating based on the XM tap selection criterion was shown to give good convergence performance for SAEC applications and has motivated the current analytical study of the convergence behavior of selective-tap algorithms.

A class of single-channel partial update algorithms employing the MMax tap-selection has recently been proposed in [12]. The main contribution of this paper is the analysis of these algorithms for time-varying unknown system conditions. In system identification applications including echo cancellation and channel equalization, such as shown in Fig. 1, an FIR adaptive filter is used to identify an unknown time-varying system that is assumed to be linear. Important performance measures for adaptive filters characterize the initial convergence rate, the residual error after convergence, the ability to track time-varying systems, and the computational complexity. This paper focuses on formulating MMax versions of the affine projection (AP) and the recursive least squares (RLS) algorithms and subsequently analyzing their performance when tracking time-varying systems. Consideration of algorithm performance under such dynamic conditions is important since, in the applications of interest, the unknown system is often continuously time-varying. It is therefore necessary to include a time-varying system model in the analysis of such adaptive algorithms as indicated in several studies including [15]–[18].

We have adopted the time-varying channel model in [19] which uses a modified first-order Markov model of the unknown system. Whereas the work in [19] specifically addresses LMS and RLS, our analysis framework extends [19] and [20] to a more general form that can be applied to a wider range of adaptive algorithms including NLMS, AP, RLS and, in particular, the MMax selective-tap algorithms that are our main focus. Through the use of this framework, this paper presents new insights into the tracking performance of selective-tap algorithms by highlighting and comparing the performances for fully-updated and MMax-based algorithms under both time-invariant and time-varying unknown system conditions. It is shown, for each algorithm, how the tracking performance is degraded by the MMax tap selection and the degradation in steady-state misalignment performance is quantified analytically under common assumptions.

This paper is organized as follows: Section II of this paper reviews the MMax-NLMS algorithm. We then extend the

MMax tap selection to the AP and RLS algorithms. Section III develops a general analysis framework for the steady-state misalignment for a time-varying case. Having established the new analysis framework and applied it to standard adaptive filtering examples, the principal contribution of Section IV is the steady-state misalignment analysis of MMax selective-tap algorithms. The analysis of MMax-NLMS includes Max-NLMS [21] as a special case. We discuss the computational complexity of the algorithms in Section V. Comparative results are shown in Section VI to verify the analytically derived misalignment performance against simulation learning curves for system identification.

## II. MMAX ADAPTIVE ALGORITHMS

The MMax-NLMS algorithm is now briefly reviewed and the concept of MMax tap selection is then extended to the AP and RLS algorithms. The main benefit reported to motivate the introduction of AP and RLS selective-tap schemes is that they are more robust to the conditioning of system identification problems with correlated inputs such as occur in SAEC [11].

### A. MMax-NLMS Algorithm

Fig. 1 shows a system identification structure in which the tap-input vector<sup>1</sup>  $\mathbf{x}_n = [x_n(0), x_n(1), \dots, x_n(L-1)]^T$  is convolved with the unknown system,  $\mathbf{h}_n = [h_n(0), h_n(1), \dots, h_n(L-1)]^T$ , to produce the signal sample  $y_n$  where the superscript  $T$  represents transposition. An adaptive filter  $\hat{\mathbf{h}}_n = [\hat{h}_n(0), \hat{h}_n(1), \dots, \hat{h}_n(L-1)]^T$  is used to identify  $\mathbf{h}_n$  by adaptively minimizing the square of the error signal

$$e_n = y_n + w_n - \hat{y}_n \quad (1)$$

where  $\hat{y}_n = \mathbf{x}_n^T \hat{\mathbf{h}}_n$  and  $w_n$  is measurement noise.

In the MMax-NLMS algorithm [22], for an adaptive filter of length  $L$ , only those taps corresponding to the  $M$  largest magnitude tap-inputs are selected for updating at each iteration. We first define the subselected tap-input vector  $\tilde{\mathbf{x}}_n = \mathbf{Q}_n \mathbf{x}_n$ , where  $\mathbf{Q}_n = \text{diag}\{\mathbf{q}_n\}$  is a  $L \times L$  tap selection matrix such that  $\mathbf{q}_n = [q_n(0), q_n(1), \dots, q_n(L-1)]^T$  and for  $i = 0, 1, \dots, L-1$

$$q_n(i) = \begin{cases} 1, & |x_n(i)| \in \{M \text{ maxima of } |\mathbf{x}_n|\} \\ 0, & \text{otherwise} \end{cases} \quad (2)$$

Defining  $\|\cdot\|_2^2$  as the squared  $l_2$ -norm, the MMax-NLMS tap-update equation is then given by

$$\hat{\mathbf{h}}_{n+1} = \hat{\mathbf{h}}_n + \mu \frac{\mathbf{Q}_n \mathbf{x}_n e_n}{\|\mathbf{x}_n\|_2^2 + \delta} \quad (3)$$

where  $\delta$  and  $\mu$  are the regularization parameter and step-size, respectively.

It has been shown in [13] that the performance of MMax-NLMS degrades only gracefully with reducing  $M$  compared to fully updated NLMS. For  $0.5L \leq M < L$ , the degradation in performance can be insignificant.

<sup>1</sup>In this work, for brevity of notation, we denote  $x(n-i)$  to be  $x_n(i)$ .

### B. MMax-RLS Algorithm

The tap-update equation of the RLS algorithm is given as [1]

$$\hat{\mathbf{h}}_{n+1} = \hat{\mathbf{h}}_n + \mathbf{k}_n e_n \quad (4)$$

where  $\mathbf{k}_n = \mathbf{R}_n^{-1} \mathbf{x}_n$  is defined as the Kalman gain such that  $\mathbf{R}_n = \sum_{j=1}^n \lambda^{n-j} \mathbf{x}_j \mathbf{x}_j^T$  is the time-averaged autocorrelation matrix with forgetting factor  $\lambda$ ,  $0 \ll \lambda < 1$ . We note that direct extension of the MMax tap selection approach achieved by sorting the magnitude of  $\mathbf{k}_n$  in (4) will not give the desired convergence behavior especially for statistically nonstationary signals such as speech. This is because the Kalman gain depends on previous values of the time-averaged autocorrelation matrix [11]. Our approach will be instead to subsample the tap-input vectors at each time iteration based on the MMax tap selection criterion. This ensures that the subselected tap-input vectors propagate consistently through the memory of the RLS algorithm.

Using  $\tilde{\mathbf{x}}_n = \mathbf{Q}_n \mathbf{x}_n$  and  $\mathbf{Q}_n$  as defined in (2), the MMax-RLS algorithm solves

$$\hat{\mathbf{h}}_n = \tilde{\mathbf{R}}_n^{-1} \tilde{\Theta}_n \quad (5)$$

where  $\tilde{\mathbf{R}}_n = \sum_{j=1}^n \lambda^{n-j} \tilde{\mathbf{x}}_j \tilde{\mathbf{x}}_j^T$ ,  $\tilde{\Theta}_n = \sum_{j=1}^n \lambda^{n-j} \tilde{\mathbf{x}}_j d_j$  and  $d_j$  is the desired signal sample at the  $j$ th iteration. The MMax-RLS tap-update equation is then given by

$$\hat{\mathbf{h}}_{n+1} = \hat{\mathbf{h}}_n + \tilde{\mathbf{k}}_n e_n \quad (6)$$

where the modified Kalman gain is

$$\tilde{\mathbf{k}}_n = \tilde{\mathbf{R}}_n^{-1} \tilde{\mathbf{x}}_n = \frac{\lambda^{-1} \tilde{\mathbf{R}}_{n-1}^{-1} \tilde{\mathbf{x}}_n}{1 + \lambda^{-1} \tilde{\mathbf{x}}_n^T \tilde{\mathbf{R}}_{n-1}^{-1} \tilde{\mathbf{x}}_n}. \quad (7)$$

Using the matrix inversion lemma [1], [23], the inversion of  $\tilde{\mathbf{R}}_n$  may then be expressed iteratively as

$$\tilde{\mathbf{R}}_n^{-1} = \frac{1}{\lambda} \left[ \tilde{\mathbf{R}}_{n-1}^{-1} - \tilde{\mathbf{k}}_n \tilde{\mathbf{x}}_n^T \tilde{\mathbf{R}}_{n-1}^{-1} \right]. \quad (8)$$

Although the Kalman gain vector  $\tilde{\mathbf{k}}_n$  is a fully populated column vector, savings in computation comes from the presence of zero elements of  $\tilde{\mathbf{R}}_{n-1}^{-1}$  and  $\tilde{\mathbf{x}}_n$  which reduces the multiplications needed to compute  $\tilde{\mathbf{R}}_{n-1}^{-1} \tilde{\mathbf{x}}_n$ . The MMax-RLS algorithm efficiently updates all taps at each iteration, and we choose to describe this as a selective-tap algorithm rather than a partial-update algorithm.

### C. MMax-AP Algorithm

The affine projection algorithm [1] incorporates multiple projections by concatenating past tap-input vectors from time iteration  $n$  to time iteration  $n - K + 1$  where  $K$  is defined as the projection order. In a similar manner, our approach for MMax-AP will be to concatenate the subselected tap-input vectors such that data propagates consistently from each iteration to

TABLE I  
MMAX TAP SELECTION

$$\begin{aligned} M &= 1, 2, \dots, L \\ \mathbf{Q}_n &= \text{diag}\{q_n(0), q_n(1), \dots, q_n(L-1)\} \\ q_n(i) &= \begin{cases} 1 & |x_n(i)| \in \{M \text{ maxima of } |\mathbf{x}_n|\} \\ 0 & \text{otherwise} \end{cases} \end{aligned}$$

TABLE II  
MMAX-NLMS

$$\begin{aligned} \hat{y}_n &= \hat{\mathbf{h}}_n^T \mathbf{x}_n \\ e_n &= d_n - \hat{y}_n \\ \hat{\mathbf{h}}_{n+1} &= \hat{\mathbf{h}}_n + \mu \frac{\mathbf{Q}_n \mathbf{x}_n e_n}{\|\mathbf{x}_n\|_2^2 + \delta} \end{aligned}$$

TABLE III  
MMAX-RLS

$$\begin{aligned} \tilde{\mathbf{x}}_n &= \mathbf{Q}_n \mathbf{x}_n \\ \tilde{\mathbf{k}}_n &= \frac{\tilde{\mathbf{R}}_{n-1}^{-1} \tilde{\mathbf{x}}_n}{\lambda + \tilde{\mathbf{x}}_n^T \tilde{\mathbf{R}}_{n-1}^{-1} \tilde{\mathbf{x}}_n} \\ \hat{y}_n &= \hat{\mathbf{h}}_n^T \mathbf{x}_n \\ e_n &= d_n - \hat{y}_n \\ \hat{\mathbf{h}}_{n+1} &= \hat{\mathbf{h}}_n + \tilde{\mathbf{k}}_n e_n \\ \tilde{\mathbf{R}}_n^{-1} &= \frac{1}{\lambda} [\tilde{\mathbf{R}}_{n-1}^{-1} - \tilde{\mathbf{k}}_n \tilde{\mathbf{x}}_n^T \tilde{\mathbf{R}}_{n-1}^{-1}] \end{aligned}$$

the next. To formulate the MMax-AP algorithm, we first define the subselected and full tap-input matrices, each of dimension  $K \times L$ , respectively, as

$$\tilde{\mathbf{X}}_n = [\tilde{\mathbf{x}}_n, \tilde{\mathbf{x}}_{n-1}, \dots, \tilde{\mathbf{x}}_{n-K+1}]^T, \quad (9)$$

$$\mathbf{X}_n = [\mathbf{x}_n, \mathbf{x}_{n-1}, \dots, \mathbf{x}_{n-K+1}]^T. \quad (10)$$

The tap-update for the MMax-AP algorithm is then given by

$$\hat{\mathbf{h}}_{n+1} = \hat{\mathbf{h}}_n + \mu \tilde{\mathbf{X}}_n^T [\mathbf{X}_n \mathbf{X}_n^T]^{-1} \mathbf{e}_n \quad (11)$$

where  $\mathbf{e}_n = [e_n, e_{n-1}, \dots, e_{n-K+1}]^T$ . Thus for  $K = 1$ , MMax-AP is equivalent to MMax-NLMS as expected. Savings in computation are achieved due to the zeros in the  $K \times L$  rectangular matrix  $\tilde{\mathbf{X}}_n$ . This reduces the number of multiplications needed to compute the term  $\tilde{\mathbf{X}}_n^T [\mathbf{X}_n \mathbf{X}_n^T]^{-1}$ . We also note that, as with MMax-RLS, MMax-AP cannot be classified as a partial-update algorithm since  $\tilde{\mathbf{X}}_n^T [\mathbf{X}_n \mathbf{X}_n^T]^{-1} \mathbf{e}_n$  is fully populated, and therefore every coefficient in  $\hat{\mathbf{h}}_n$  will be updated at each iteration. Consequently, we describe MMax-AP also as a selective-tap algorithm.

The MMax-NLMS, MMax-RLS, and MMax-AP algorithms are summarized in Tables I–IV, and we present simulation results to illustrate their operation in Section VI.

TABLE IV  
MMAX-AP

$\mathbf{X}_n$	$= [\mathbf{x}_n, \mathbf{x}_{n-1}, \dots, \mathbf{x}_{n-K+1}]^T$
$\tilde{\mathbf{x}}_n$	$= \mathbf{Q}_n \mathbf{x}_n$
$\tilde{\mathbf{X}}_n$	$= [\tilde{\mathbf{x}}_n, \tilde{\mathbf{x}}_{n-1}, \dots, \tilde{\mathbf{x}}_{n-K+1}]^T$
$\mathbf{d}_n$	$= [d_n, d_{n-1}, \dots, d_{n-K+1}]^T$
$\hat{\mathbf{y}}_n$	$= \mathbf{X}_n \hat{\mathbf{h}}_n$
$\mathbf{e}_n$	$= \mathbf{d}_n - \hat{\mathbf{y}}_n$
$\hat{\mathbf{h}}_{n+1}$	$= \hat{\mathbf{h}}_n + \mu \tilde{\mathbf{X}}_n^T [\mathbf{X}_n \mathbf{X}_n^T]^{-1} \mathbf{e}_n$

TABLE V  
PROJECTION ORDER AND  $\Gamma_{n-j}$  FOR VARIOUS ALGORITHMS

Algorithm	Projection order	$\Gamma_{n-j}$
LMS	$K = 1$	$2\mu \mathbf{I}$
NLMS	$K = 1$	$\frac{2\mu}{\mathbf{x}_n^T \mathbf{x}_n} \mathbf{I}$
AP	$K$	$\frac{2\mu}{\mathbf{x}_{n-j}^T \mathbf{x}_{n-j}} \mathbf{I}$
RLS	$K = 1$	$\mathbf{R}_n^{-1}$

### III. GENERAL MISALIGNMENT ANALYSIS FOR TIME VARYING SYSTEMS

We consider adaptive algorithms of the form

$$\hat{\mathbf{h}}_{n+1} = \hat{\mathbf{h}}_n + \sum_{j=0}^{K-1} \Gamma_{n-j} \mathbf{x}_{n-j} e_{n-j} \quad (12)$$

where  $\mathbf{x}_{n-j} = [x_{n-j}(0), x_{n-j}(1), \dots, x_{n-j}(L-1)]^T$  is the tap-input vector, which is assumed to be real-valued and  $e_{n-j}$  represents the error signal. The  $L \times L$  adaptation control matrix  $\Gamma_{n-j}$  is defined in Table V for the respective algorithms.

Note that for the AP algorithm, we have assumed, similar to [24], [25], that for  $1 \leq K \ll L$ ,  $\mathbf{X}_n \mathbf{X}_n^T = \text{diag}\{\|\mathbf{x}_n\|_2^2, \|\mathbf{x}_{n-1}\|_2^2, \dots, \|\mathbf{x}_{n-K+1}\|_2^2\}$  such that

$$[\mathbf{X}_n \mathbf{X}_n^T]^{-1} = \begin{bmatrix} \frac{1}{\|\mathbf{x}_n\|_2^2} & 0 & \dots & 0 \\ 0 & \frac{1}{\|\mathbf{x}_{n-1}\|_2^2} & \ddots & 0 \\ \vdots & \ddots & \ddots & 0 \\ 0 & \dots & 0 & \frac{1}{\|\mathbf{x}_{n-K+1}\|_2^2} \end{bmatrix}. \quad (13)$$

#### A. Time-Varying System Model

The modified first-order Markov model [19] is employed to represent a time-varying unknown system

$$\mathbf{h}_{n+1} = \xi \mathbf{h}_n + \sqrt{1 - \xi^2} \mathbf{s}_n \quad (14)$$

where  $\mathbf{h}_n$  is the impulse response of the unknown system, and  $\mathbf{s}_n = [s_n(0), s_n(1), \dots, s_n(L-1)]^T$  is a noise process drawn from the normal distribution  $\mathcal{N}(0, \sigma_s^2)$ . As shown in [19], this

model has the key features that 1) the single parameter  $0 \ll \xi < 1$  controls the relative contributions to the instantaneous values of the coefficients of “system memory” (the term  $\xi \mathbf{h}_n$ ) and “innovations” (the term  $\sqrt{1 - \xi^2} \mathbf{s}_n$ ), 2) the average power of the norm of the coefficients is independent of  $\xi$ . It is subsequently shown in [19] that the system variation, which notionally indicates the difficulty of tracking by an adaptive filter, is a monotonic decreasing function of  $\xi$ . Defining  $\sigma_s^2$  as the variance of  $\mathbf{s}_n$ , it has been found experimentally, that  $\xi = 0.9999$  for the Markov model with  $\sigma_s^2 = 1$  given in (14) is comparatively equivalent to a source moving at  $0.2 \text{ ms}^{-1}$  for acoustic impulse responses  $\mathbf{h}_n$  generated using the method of images [26].

For the purpose of this analysis, defining  $E\{\cdot\}$  as the expectation operator, we first assume  $E\{\mathbf{h}_n\} = 0$ ,  $E\{w_n\} = 0$  and that  $\mathbf{h}_n$  and  $w_n$  are independent.<sup>2</sup> We also assume that the dimension of  $\hat{\mathbf{h}}_n$  has been chosen to match the dimension of  $\mathbf{h}_n$ . We then define the misalignment vector

$$\mathbf{v}_n = \hat{\mathbf{h}}_n - \mathbf{h}_n \quad (15)$$

which results in the error signal given by

$$e_n = w_n - \mathbf{x}_n^T \mathbf{v}_n. \quad (16)$$

Using (12) and (14)–(16), we obtain

$$\begin{aligned} \mathbf{v}_{n+1} &= \hat{\mathbf{h}}_{n+1} - \mathbf{h}_{n+1} \\ &= \hat{\mathbf{h}}_n - \xi \mathbf{h}_n - \sqrt{1 - \xi^2} \mathbf{s}_n + \sum_{j=0}^{K-1} \Gamma_{n-j} \mathbf{x}_{n-j} e_{n-j} \\ &= \mathbf{v}_n + (1 - \xi) \mathbf{h}_n + \sum_{j=0}^{K-1} \Gamma_{n-j} \mathbf{x}_{n-j} w_{n-j} \\ &\quad - \sqrt{1 - \xi^2} \mathbf{s}_n - \sum_{j=0}^{K-1} \Gamma_{n-j} \mathbf{x}_{n-j} \mathbf{x}_{n-j}^T \mathbf{v}_{n-j} \end{aligned} \quad (17)$$

from which

$$\begin{aligned} \mathbf{R}_{\mathbf{v}, n+1} &= E\{\mathbf{v}_{n+1} \mathbf{v}_{n+1}^T\} \\ &= \mathbf{R}_{\mathbf{v}, n} + 2(1 - \xi) \sigma_s^2 \mathbf{I} \\ &\quad + K \sigma_w^2 \sum_{j=0}^{K-1} E\{\Gamma_{n-j} \mathbf{x}_{n-j} \mathbf{x}_{n-j}^T \Gamma_{n-j}^T\} \\ &\quad - \mathbf{R}_{\mathbf{v}, n} \sum_{j=0}^{K-1} E\{\Gamma_{n-j} \mathbf{x}_{n-j} \mathbf{x}_{n-j}^T\} \\ &\quad - \mathbf{R}_{\mathbf{v}, n} \sum_{j=0}^{K-1} E\{\mathbf{x}_{n-j} \mathbf{x}_{n-j}^T \Gamma_{n-j}^T\} \\ &\quad + E\left\{ \sum_{j=0}^{K-1} \Gamma_{n-j} \mathbf{x}_{n-j} \mathbf{x}_{n-j}^T \mathbf{v}_{n-j} \right. \\ &\quad \times \left. \sum_{i=0}^{K-1} \mathbf{v}_{n-i}^T \mathbf{x}_{n-i} \mathbf{x}_{n-i}^T \Gamma_{n-i}^T \right\} \end{aligned} \quad (18)$$

<sup>2</sup>If  $E\{\mathbf{h}_n\} \neq 0$ , a bias will be induced in the variance of  $\mathbf{h}_n$  by an amount proportional to the square values of  $E\{\mathbf{h}_n\}$ .

where  $\mathbf{I}$  is the identity matrix and we have made use of the following relations

$$\begin{aligned} E\{\mathbf{v}_n \mathbf{v}_n^T\} &= \mathbf{R}_{\mathbf{v},n}, \\ E\{w_{n-i} w_{n-j}\} &= \begin{cases} \sigma_w^2 & i = j \\ 0 & \text{otherwise} \end{cases}, \\ E\left\{\mathbf{v}_n \sum_{j=0}^{K-1} \mathbf{v}_{n-j}^T \mathbf{x}_{n-j} \mathbf{x}_{n-j}^T \Gamma_{n-j}^T\right\} \\ &= \mathbf{R}_{\mathbf{v},n} \sum_{j=0}^{K-1} E\{\mathbf{x}_{n-j} \mathbf{x}_{n-j}^T \Gamma_{n-j}^T\} \end{aligned}$$

and, from the definition of the first-order Markov model as shown in (14)

$$E\{\mathbf{h}_n \mathbf{h}_n^T\} = E\{\mathbf{s}_n \mathbf{s}_n^T\} = \sigma_s^2 \mathbf{I}. \quad (19)$$

Following the approach adopted in [1], we assume that the time variations of  $\mathbf{h}_n$  are sufficiently slow that the adaptive filter is able to track the unknown system to within a time lag, and that after convergence,  $\mathbf{v}_{n-j}$  is fluctuating around its mean  $\forall n, j$  so that  $E\{\mathbf{v}_{n-j} \mathbf{v}_{n-j}^T\} \approx E\{\mathbf{v}_n \mathbf{v}_n^T\} = \mathbf{R}_{\mathbf{v},n}$ . We define  $\mathbf{R}_{\mathbf{v}}$  as autocorrelation matrix of the mean weight error vector which is approximately time-invariant under these assumptions. Defining the normalized misalignment as

$$\eta = \frac{\|\mathbf{h}_n - \hat{\mathbf{h}}_n\|_2^2}{\|\mathbf{h}_n\|_2^2} = \frac{\eta'}{\|\mathbf{h}_n\|_2^2} \quad (20)$$

the steady-state misalignment can be expressed as  $\eta' = \text{tr}\{\mathbf{R}_{\mathbf{v}}\}$  where  $\text{tr}\{\cdot\}$  is the trace operator.

### B. Steady-State Misalignment for $K = 1, M = L$

We initially consider a fully updated algorithm such that  $\Gamma_n = \Gamma, \forall n$ , is time-invariant, and statistically stationary inputs  $\mathbf{x}_n$ . Using the factorization property of independent Gaussian variables [1] as shown in Appendix I and denoting  $\mathbf{R}_{\mathbf{x}}$  as the autocorrelation matrix of the input signal, the expectations in (18) can be evaluated for  $K = 1$  using

$$\begin{aligned} E\{\Gamma_n \mathbf{x}_n \mathbf{x}_n^T \Gamma_n^T\} &= \Gamma \mathbf{R}_{\mathbf{x}} \Gamma \\ E\{\Gamma_n \mathbf{x}_n \mathbf{x}_n^T\} &= \Gamma \mathbf{R}_{\mathbf{x}} \\ E\{\mathbf{x}_n \mathbf{x}_n^T \Gamma_n^T\} &= \mathbf{R}_{\mathbf{x}} \Gamma \\ E\{\Gamma_n \mathbf{x}_n \mathbf{x}_n^T \mathbf{v}_n \mathbf{v}_n^T \mathbf{x}_n \mathbf{x}_n^T \Gamma_n^T\} &= \Gamma [2\mathbf{R}_{\mathbf{x}} \mathbf{R}_{\mathbf{v},n} \mathbf{R}_{\mathbf{x}} \\ &\quad + \mathbf{R}_{\mathbf{x}} \text{tr}\{\mathbf{R}_{\mathbf{x}} \mathbf{R}_{\mathbf{v},n}\}] \Gamma. \end{aligned} \quad (21)$$

Substituting (19) and (21) into (18) gives

$$\begin{aligned} \mathbf{R}_{\mathbf{v},n+1} &= \mathbf{R}_{\mathbf{v},n} - \mathbf{R}_{\mathbf{v},n} \Gamma \mathbf{R}_{\mathbf{x}} - \mathbf{R}_{\mathbf{v},n} \mathbf{R}_{\mathbf{x}} \Gamma \\ &\quad + \Gamma [2\mathbf{R}_{\mathbf{x}} \mathbf{R}_{\mathbf{v},n} \mathbf{R}_{\mathbf{x}} + \mathbf{R}_{\mathbf{x}} \text{tr}\{\mathbf{R}_{\mathbf{x}} \mathbf{R}_{\mathbf{v},n}\}] \Gamma \\ &\quad + \Gamma \mathbf{R}_{\mathbf{x}} \Gamma \sigma_w^2 + 2(1 - \xi) \sigma_s^2 \mathbf{I}. \end{aligned} \quad (22)$$

We proceed by considering  $\Gamma = c\mathbf{I}$  and Gaussian input with variance  $\sigma_x^2$  giving  $\mathbf{R}_{\mathbf{x}} = \sigma_x^2 \mathbf{I}$ , where  $c$  is an algorithmic dependent term. Assuming  $\mathbf{v}_n$  is fluctuating around its mean after convergence giving  $\mathbf{R}_{\mathbf{v},n} \approx \mathbf{R}_{\mathbf{v}}$ , where  $\mathbf{R}_{\mathbf{v}}$  is the autocorrelation matrix of the mean weight error vector, the steady-state misalignment is then given by  $\eta' = \text{tr}\{\mathbf{R}_{\mathbf{v}}\} \approx \text{tr}\{\mathbf{R}_{\mathbf{v},n}\}$ . Hence the trace of (22) can be simplified giving

$$\begin{aligned} \eta' &= \eta' - 2c\sigma_x^2 \eta' + 2\sigma_x^4 c^2 \eta' + \sigma_x^4 c^2 L \eta' \\ &\quad + c^2 \sigma_x^2 \sigma_w^2 L + 2(1 - \xi) \sigma_s^2 L \end{aligned}$$

from which we can then express

$$\eta' = \frac{c\sigma_w^2 L}{2\phi} + \frac{(1 - \xi)L\sigma_s^2}{c\sigma_x^2 \phi} \quad (23)$$

where

$$\phi = 1 - c\sigma_x^2 \left(1 + \frac{L}{2}\right) \quad (24)$$

and  $c$  is an algorithm dependent term. Adopting the terminology of [18], the first term in (23) corresponds to the *estimation variance* and is dependent on measurement noise  $w_n$ , and the second term in (23) corresponds to the *lag variance* and is due to system time variation  $\xi$ . We note from (23) that these two terms are uncoupled.

For the LMS case,  $c = 2\mu$  and hence

$$\eta'_{\text{LMS}} = \frac{\mu\sigma_w^2 L}{\phi} + \frac{(1 - \xi)L\sigma_s^2}{2\mu\sigma_x^2 \phi}. \quad (25)$$

The estimation variance term of this result is, as expected, proportional to  $\mu$  and consistent with that presented in [1] for which it is assumed  $\phi \approx 1$ . However, the analysis presented here needs no such assumption. The lag variance term is inversely proportional to  $\mu$  and linearly dependent on the system variation parameter  $\xi$ .

For NLMS,  $c = 2\mu/(L\sigma_x^2)$  and

$$\eta'_{\text{NLMS}} = \frac{\mu\sigma_w^2}{\sigma_x^2 \phi} + \frac{(1 - \xi)L^2 \sigma_s^2}{2\mu\phi}. \quad (26)$$

It is interesting, from a step-size control point of view, that we proceed to evaluate the step-size  $\mu_{\text{mis}}$  which achieves the lowest misalignment by letting

$$\gamma = 2(1 + L/2)/L \quad (27)$$

and differentiating (26) with respect to  $\mu$  to obtain

$$\begin{aligned} \frac{d\eta'_{\text{NLMS}}}{d\mu} &= \frac{\sigma_w^2}{\sigma_x^2} \left[ \frac{\gamma\mu}{(1 - \gamma\mu)^2} + \frac{1}{1 - \gamma\mu} \right] \\ &\quad + \frac{(1 - \xi)L^2 \sigma_s^2}{2} \left[ \frac{2\gamma\mu - 1}{\mu^2(1 - \gamma\mu)^2} \right]. \end{aligned} \quad (28)$$

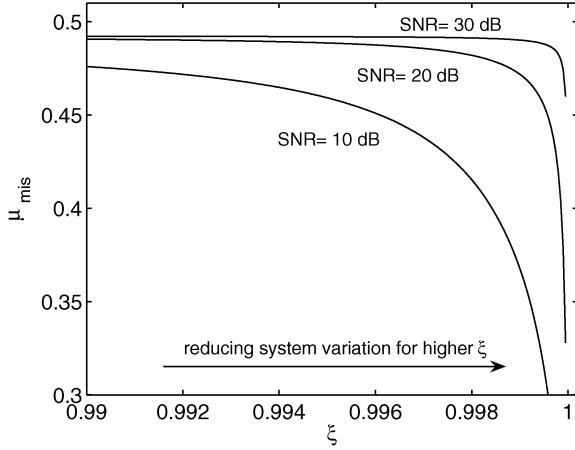


Fig. 2. Variation of  $\mu_{\text{mis}}$  with  $\xi$  under various SNR conditions.

Setting  $d\eta'_{\text{NLMS}}/d\mu = 0$ , we obtain a quadratic equation in terms of  $\mu_{\text{mis}}$ . Under the condition that  $0 < \mu_{\text{mis}} \leq 1$ , we may solve for  $\mu_{\text{mis}}$  as

$$\mu_{\text{mis}} = 0.5 \frac{\sigma_x^2}{\sigma_w^2} \left[ - (1 - \xi) L^2 \sigma_s^2 \gamma + \sqrt{[(1 - \xi) L^2 \sigma_s^2 \gamma]^2 + 2 \left( \frac{\sigma_w^2}{\sigma_x^2} \right) (1 - \xi) L^2 \sigma_s^2} \right]. \quad (29)$$

Fig. 2 illustrates the variation of  $\mu_{\text{mis}}$  with  $\xi$  under various signal-to-noise ratio (SNR) conditions. The SNR was computed using  $w_n$  and  $y_n$  where the latter is obtained by convolving  $x_n$  with  $\mathbf{h}_n$  as shown in Fig. 1. The parameters for this simulation are  $L = 128$ ,  $\sigma_s^2 = 1$ , and  $\sigma_x^2 = 0.962$ . We may now see, for each case of SNR, the well-known result that as  $\xi \rightarrow 1$ ,  $\mu_{\text{mis}} \rightarrow 0$  and hence a smaller step-size achieves a lower steady-state misalignment, though at the expense of reduced convergence rate. Under time-varying unknown system conditions  $\xi < 1$ , it can be seen as expected that  $\mu_{\text{mis}}$  increases smoothly as  $\xi$  reduces, since for higher time-varying unknown system condition,  $\mu$  must be sufficiently high for tracking. We also note that for any given  $\xi$ ,  $\mu_{\text{mis}}$  increases with SNR. As will be seen through simulation examples in Section VI-D, under the condition  $\xi < 1$ , performance of NLMS in terms of convergence rate and steady-state misalignment increases with  $\mu$  for  $0 < \mu \leq \mu_{\text{mis}}$ .

For RLS, we can determine  $c$  in

$$\Gamma = c\mathbf{I} = \mathbf{R}_n^{-1} \quad (30)$$

by considering the exponentially weighted recursive realization of the autocorrelation matrix, from which we recall the definition from Section II-B,  $\mathbf{R}_n = \sum_{j=1}^n \lambda^{n-j} \mathbf{x}_j \mathbf{x}_j^T$ . In the limit  $n \rightarrow \infty$

$$\begin{aligned} E \left\{ \lim_{n \rightarrow \infty} \mathbf{R}_n \right\} &= \lim_{n \rightarrow \infty} (\lambda^{n-1} + \lambda^{n-2} + \dots + 1) \mathbf{R}_x \\ &= \frac{1}{1 - \lambda} \mathbf{R}_x \end{aligned} \quad (31)$$

where  $\mathbf{R}_x$  is the true input signal autocorrelation matrix. Using a quasi-deterministic approximation for large  $n$  [1], we obtain  $\mathbf{R}_n \approx \mathbf{R}_x / (1 - \lambda)$  and hence

$$\Gamma = \mathbf{R}_n^{-1} \approx (1 - \lambda) \mathbf{R}_x^{-1}. \quad (32)$$

For the case when the input signal  $x_n$  is drawn from an independent and identically distributed (i.i.d.) process such that  $E\{x_n\} = 0$  with variance  $\sigma_x^2$ , then  $c = (1 - \lambda) / \sigma_x^2$  and using (23), the steady-state misalignment is given as

$$\eta'_{\text{RLS}} \approx \frac{(1 - \lambda) L \sigma_w^2}{2 \sigma_x^2 \phi} + \frac{(1 - \xi) L \sigma_s^2}{(1 - \lambda) \phi} \quad (33)$$

where the term  $\phi$  is defined in (24). We may now see the well-known result that for a time-invariant unknown system condition  $\xi = 1$ , we have

$$\frac{d\eta'_{\text{RLS}}}{d\lambda} = - \frac{L \sigma_w^2}{2 \sigma_x^2 \phi^2} < 0 \quad (34)$$

and hence the steady-state misalignment is a decreasing function of  $\lambda$ , although for smaller  $\lambda$  the rate of convergence is increased.

We can analyze the effect of  $\xi$  on  $\eta'_{\text{RLS}}$  by first differentiating  $\eta'_{\text{RLS}}$  with respect to  $\xi$  and finding the boundary condition of  $\lambda$ . We first assume that

$$d\eta'_{\text{RLS}}/d\xi < 0 \quad (35)$$

and noting that  $L \sigma_s^2 > 0$  and  $1 - \lambda > 0$ , we obtain the condition

$$\lambda > 1 - [1 / (1 + L/2)]. \quad (36)$$

For a typical range [27],  $1 - 1/(3L) \leq \lambda \leq 1 - 1/(10L)$  and since  $L > 1$ , it can be seen that the conditions (36) and consequently (35) are satisfied. Hence,  $\eta'_{\text{RLS}}$  reduces as  $\xi \rightarrow 1$ .

### C. Steady-State Misalignment for $K \neq 1$ , $M = L$

This analysis can be applied to the AP algorithm for the condition  $1 \leq K \ll L$ . With reference to Table V, we assume that the input signal is drawn from an i.i.d process such that  $\mathbf{x}_{n-j}^T \mathbf{x}_{n-j} = L \sigma_x^2$ ,  $\forall n, j$ , hence giving  $\Gamma_{n-j} = \Gamma$ . Exploiting the linear property of the expectation operator, we may then evaluate the terms in (18) using the following relations:

$$\begin{aligned} \sum_{j=0}^{K-1} E \{ \Gamma_{n-j} \mathbf{x}_{n-j} \mathbf{x}_{n-j}^T \Gamma_{n-j}^T \} &= K \Gamma \mathbf{R}_x \Gamma \\ \sum_{j=0}^{K-1} E \{ \Gamma_{n-j} \mathbf{x}_{n-j} \mathbf{x}_{n-j}^T \} &= K \Gamma \mathbf{R}_x \\ \sum_{j=0}^{K-1} E \{ \mathbf{x}_{n-j} \mathbf{x}_{n-j}^T \Gamma_{n-j}^T \} &= K \mathbf{R}_x \Gamma. \end{aligned} \quad (37)$$

Following a similar approach to (21), where we have used the factorization property of independent Gaussian variables as

shown in Appendix I, the last term of (18) can be simplified as follows:

$$\begin{aligned} & E \left\{ \sum_{j=0}^{K-1} \Gamma_{n-j} \mathbf{x}_{n-j} \mathbf{x}_{n-j}^T \mathbf{v}_{n-j} \sum_{i=0}^{K-1} \mathbf{v}_{n-i}^T \mathbf{x}_{n-i} \mathbf{x}_{n-i}^T \Gamma_{n-i}^T \right\} \\ &= E \left\{ \sum_{j=0}^{K-1} \sum_{i=0}^{K-1} \Gamma_{n-j} \mathbf{x}_{n-j} \mathbf{x}_{n-j}^T \mathbf{v}_{n-j} \mathbf{v}_{n-i}^T \mathbf{x}_{n-i} \mathbf{x}_{n-i}^T \Gamma_{n-i}^T \right\} \\ &= K^2 \Gamma [2\mathbf{R}_x \mathbf{R}_{v,n} \mathbf{R}_x + \mathbf{R}_x \text{tr}\{\mathbf{R}_x \mathbf{R}_{v,n}\}] \Gamma. \end{aligned} \quad (38)$$

Substituting (37) and (38) into (18), we obtain

$$\begin{aligned} \mathbf{R}_{v,n+1} &= \mathbf{R}_{v,n} + 2(1-\xi)\sigma_s^2 \mathbf{I} + K^2 \sigma_w^2 \Gamma \mathbf{R}_x \Gamma^T \\ &\quad - K \mathbf{R}_{v,n} \Gamma \mathbf{R}_x - K \mathbf{R}_{v,n} \mathbf{R}_x \Gamma \\ &\quad + K^2 \Gamma [2\mathbf{R}_x \mathbf{R}_{v,n} \mathbf{R}_x + \mathbf{R}_x \text{tr}\{\mathbf{R}_x \mathbf{R}_{v,n}\}] \Gamma. \end{aligned} \quad (39)$$

Similar to (22), we assume that  $\mathbf{R}_{v,n}$  is fluctuating around its mean,  $\Gamma = c\mathbf{I}$  and  $\mathbf{R}_x = \sigma_x^2 \mathbf{I}$ . We can then simplify (39) and using  $c = 2\mu/(L\sigma_x^2)$ , we write the steady-state misalignment as

$$\eta'_{\text{AP}} = \frac{K\mu\sigma_w^2}{\sigma_x^2\phi_a} + \frac{(1-\xi)\sigma_s^2 L^2}{2\mu K\phi_a} \quad (40)$$

where

$$\phi_a = 1 - Kc\sigma_x^2 \left(1 + \frac{L}{2}\right). \quad (41)$$

We note that (41) is similar to (24) except for a projection order term  $K$ . Furthermore, when  $K = 1$ ,  $\phi_a$  reduces to  $\phi$  giving  $\eta'_{\text{AP}} = \eta'_{\text{NLMS}}$  as expected.

#### IV. MISALIGNMENT ANALYSIS OF ALGORITHMS EMPLOYING MMAX TAP SELECTION

##### A. Misalignment Analysis of MMax-NLMS

Partial-update NLMS algorithms have been analyzed in, for example, [2]–[4], [21], [28]. The MMax-NLMS [22] algorithm is characterized by (12) for  $K = 1$  with  $\Gamma_n = \mu_n \mathbf{Q}_n$  in which the elements of the diagonal matrix  $\mathbf{Q}_n = \text{diag}\{\mathbf{q}_n\}$  are determined from (2) and  $\tilde{\mathbf{x}}_n = \mathbf{Q}_n \mathbf{x}_n$  is the subselected tap-input vector.

Employing a contraction mapping approach [29], convergence in the mean can be shown when

$$0 < \|E\{\mathbf{I} - \mu_n \mathbf{Q}_n \mathbf{x}_n \mathbf{x}_n^T\}\|_2 < 1 \quad (42)$$

which implies for i.i.d.  $x_n$

$$0 < \mu_n < \frac{2}{\sum_{i=0}^{L-1} q_n(i) x_n^2(i)}. \quad (43)$$

For convergence in the mean square, we start by considering (18) and the evaluation of  $E\{\Gamma_n \mathbf{x}_n \mathbf{x}_n^T\}$ . We note that elements  $q_n(i)$ ,  $i = 1, \dots, L$ , are not independent of  $x_n(i)$  as they ensure

that only the  $M$  largest  $|x_n(i)|$  are selected. The  $M$  selected samples are assumed to have zero mean and exploiting the mean ergodic theorem [1], the variance of  $\tilde{\mathbf{x}}_n$  is

$$\tilde{\sigma}_x^2 = \frac{1}{L} \sum_{i=0}^{L-1} \tilde{x}_n^2(i). \quad (44)$$

Assuming that  $\mathbf{x}_n \mathbf{x}_n^T$  is diagonal and using  $E\{\mu_n\} = c$ , a scalar constant, we can evaluate

$$\begin{aligned} E\{\Gamma_n \mathbf{x}_n \mathbf{x}_n^T\} &= E\{\mathbf{x}_n \mathbf{x}_n^T \Gamma_n^T\} = E\{\mu_n\} E\{\mathbf{Q}_n \mathbf{x}_n \mathbf{x}_n^T\} \\ &= \frac{M}{L} c \tilde{\sigma}_x^2 \mathbf{I}. \end{aligned} \quad (45)$$

The condition  $E\{\Gamma_n\} = \Gamma$  implicit in (21) is not valid in this case. However, we can proceed to evaluate  $\text{tr}\{\mathbf{R}_{v,n}\}$  using

$$\begin{aligned} & \text{tr}\{E\{\Gamma_n \mathbf{x}_n \mathbf{x}_n^T \mathbf{v}_n \mathbf{v}_n^T \mathbf{x}_n \mathbf{x}_n^T \Gamma_n^T\}\} \\ &= \text{tr}\{c^2 E\{\mathbf{Q}_n \mathbf{x}_n \mathbf{x}_n^T \mathbf{v}_n \mathbf{v}_n^T \mathbf{x}_n \mathbf{x}_n^T\}\} \\ &= c^2 \text{tr}\left\{\mathbf{R}_{v,n}(L+2) \frac{M}{L} \tilde{\sigma}_x^2 \sigma_x^2 \mathbf{I}\right\} \\ &= c^2 \text{tr}\{\mathbf{R}_{v,n}\}(L+2) \frac{M}{L} \tilde{\sigma}_x^2 \sigma_x^2, \\ & \text{tr}\{E\{\Gamma_n \mathbf{x}_n \mathbf{x}_n^T \Gamma_n^T\}\} \\ &= \text{tr}\left\{\frac{M}{L} c^2 \tilde{\sigma}_x^2 \mathbf{I}\right\} \\ &= \frac{M}{L} c^2 \tilde{\sigma}_x^2 L. \end{aligned} \quad (46)$$

Substituting (45), (46), and (19) into (18) and letting  $K = 1$

$$\begin{aligned} \text{tr}\{\mathbf{R}_{v,n+1}\} &= \text{tr}\{\mathbf{R}_{v,n}\} - 2\text{tr}\{\mathbf{R}_{v,n}\} \frac{M}{L} c \tilde{\sigma}_x^2 \\ &\quad + c^2 \text{tr}\{\mathbf{R}_{v,n}\}(L+2) \frac{M}{L} \tilde{\sigma}_x^2 \sigma_x^2 \\ &\quad + \frac{M}{L} c^2 \tilde{\sigma}_x^2 \sigma_w^2 L + 2(1-\xi)\sigma_s^2 L \\ &= \text{tr}\{\mathbf{R}_{v,n}\} \left[1 - 2\frac{M}{L} c \tilde{\sigma}_x^2 \right. \\ &\quad \left. + (L+2) \frac{M}{L} c^2 \tilde{\sigma}_x^2 \sigma_x^2 \right] \\ &\quad + M c^2 \tilde{\sigma}_x^2 \sigma_w^2 + 2(1-\xi)L\sigma_s^2. \end{aligned} \quad (47)$$

Applying the contraction mapping concept we can show convergence when

$$T = \left|1 - 2\frac{M}{L} c \tilde{\sigma}_x^2 + (L+2) \frac{M}{L} c^2 \tilde{\sigma}_x^2 \sigma_x^2\right| < 1 \quad (48)$$

and the convergence speed is faster for smaller values of  $T$ . It can therefore be seen from (48) that, for typical values of  $c \ll 1$ , maximum convergence speed will be when  $M = L$ . Therefore, MMax-NLMS suffers a decrease in convergence speed proportionate to  $M/L$  as compared to NLMS. For the particular case of MMax-NLMS,  $c = 2\mu/(L\sigma_x^2)$  and we obtain the condition

$$\left|1 - 4\mu \frac{M \tilde{\sigma}_x^2}{L^2 \sigma_x^2} + 4\mu^2 (L+2) M \tilde{\sigma}_x^2 L^3 \sigma_x^2\right| < 1 \quad (49)$$

so that

$$0 < \mu < \frac{L}{L+2}. \quad (50)$$

We note that our analysis will not hold for an example case of a ramped input signal,  $x_n(i) > x_n(i+1)$ ,  $\forall n$ , for which tap-indices  $i = M+1, \dots, L$  will never be selected for updating by MMax-NLMS. However, such cases are unlikely to occur in practice.

The misalignment for MMax-NLMS can be found from (47) and using the approach of (23) as

$$\begin{aligned} \text{tr}\{\mathbf{R}_v\} & \left[ 2\frac{M}{L}c\tilde{\sigma}_x^2 - (L+2)\frac{M}{L}c^2\tilde{\sigma}_x^2\sigma_x^2 \right] \\ & = M c^2 \tilde{\sigma}_x^2 \sigma_w^2 + 2(1-\xi)L\sigma_s^2 \end{aligned}$$

resulting in

$$\text{tr}\{\mathbf{R}_v\} = \frac{c\sigma_w^2 L}{2 - (L+2)c\sigma_x^2} + \frac{2(1-\xi)L^2\sigma_s^2/M}{2c\tilde{\sigma}_x^2 - (L+2)c^2\tilde{\sigma}_x^2\sigma_x^2}. \quad (51)$$

For MMax-NLMS where  $c = 2\mu/(L\sigma_x^2)$ , the steady-state misalignment is then

$$\eta'_{\text{MMax-NLMS}} = \frac{\mu\sigma_w^2}{\sigma_x^2\phi} + \frac{L\sigma_x^2}{\tilde{\sigma}_x^2 M} \frac{(1-\xi)L^2\sigma_s^2}{2\mu\phi} \quad (52)$$

where the term  $\phi$  is defined in (24).

We first note that the *estimation variance* term is identical to that of NLMS. Thus, for a time-invariant system with  $\xi = 1$ ,  $\eta'_{\text{MMax-NLMS}} = \eta'_{\text{NLMS}}$ . Comparing (52) with (26), we next note that an additional factor arises in the lag variance of  $L\sigma_x^2/(\tilde{\sigma}_x^2 M)$  for MMax-NLMS compared to NLMS. To quantify the closeness of tap selection to that of a full tap-input vector in an MMax sense, we employ the M-ratio [11]

$$\mathcal{M} = \|\mathbf{Q}_n \mathbf{x}_n\|_2^2 / \|\mathbf{x}_n\|_2^2 = \tilde{\sigma}_x^2 / \sigma_x^2. \quad (53)$$

As shown in [11],  $\mathcal{M}$  exhibits only a modest reduction for  $0.5L \leq M < L$  and hence a graceful reduction in convergence rate is expected over this range of  $M$  as compared to fully updated NLMS. We now analyze the steady-state misalignment due to tap selection  $M$  by first noting from (52) that, under a time-varying unknown system condition  $\xi < 1$ , the lag variance is proportional to the term

$$\psi = L\sigma_x^2 / (\tilde{\sigma}_x^2 M) = L/(M\mathcal{M}). \quad (54)$$

Fig. 3, shows the variation of  $\psi$  with the number of selected taps  $M$  for  $L = 128$  using a zero mean, unit variance white Gaussian noise (WGN) input sequence. We note that for  $M = L$ ,  $\psi = 1$  since  $\tilde{\sigma}_x^2 = \sigma_x^2$  and  $\mathcal{M} = 1$ . More importantly,  $\psi$  increases smoothly with reducing  $M$  within the region  $0.5L \leq M < L$  and hence, for this range of  $M$  with reduced computational complexity, we would expect only a graceful degradation in steady-state misalignment performance for the time-varying case  $\xi < 1$ .

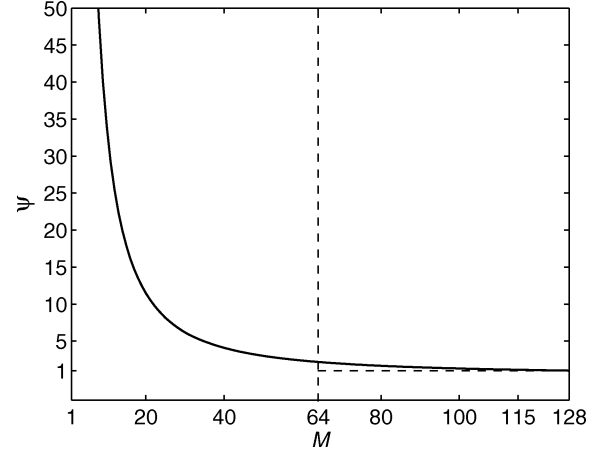


Fig. 3. Variation of  $\psi$  with  $M$  for MMax-NLMS with  $\xi < 1$  showing a modest increment in  $\psi$  for  $0.5L \leq M < L$ .

Similar to the NLMS algorithm discussed in Section III-B, it is of interest that we evaluate the step-size which achieves the lowest misalignment  $\mu_{\text{mis}}$  by first writing for clarity

$$\gamma = 2(1 + L/2)/L. \quad (55)$$

Differentiating (52) with respect to  $\mu$  and solving the quadratic equation for  $\mu_{\text{mis}}$ , we obtain for MMax-NLMS,

$$\begin{aligned} \mu_{\text{mis}} &= 0.5 \frac{\sigma_x^2}{\sigma_w^2} \left[ -\psi(1-\xi)L^2\sigma_s^2\gamma \right. \\ & \quad \left. + \sqrt{[\psi(1-\xi)L^2\sigma_s^2\gamma]^2 + 2\psi \left( \frac{\sigma_w^2}{\sigma_x^2} \right) (1-\xi)L^2\sigma_s^2} \right]. \quad (56) \end{aligned}$$

As we shall see from simulations in Section VI-D, if  $\mu_{\text{mis}} < \mu \leq 1$  under the condition  $\xi < 1$ , the convergence rate increases with  $\mu$  but at the expense of poorer steady-state misalignment. Consequently, for MMax-NLMS, the optimal  $\mu$  giving the highest rate of convergence while satisfying the minimum misalignment under time-varying conditions is  $\mu_{\text{mis}}$  given in (56).

### B. Misalignment Analysis of MMax-RLS

Using (6) and (7), the tap update equation for the MMax-RLS algorithm may be expressed as

$$\hat{\mathbf{h}}_{n+1} = \hat{\mathbf{h}}_n + \tilde{\mathbf{R}}_n^{-1} \mathbf{Q}_n \mathbf{x}_n e_n. \quad (57)$$

In this case, we have  $K = 1$  in the general formulation of (12) and

$$\Gamma_n = \tilde{\mathbf{R}}_n^{-1} \mathbf{Q}_n \quad (58)$$

in (12) where

$$\tilde{\mathbf{R}}_n = \sum_{j=1}^n \lambda^{n-j} \mathbf{Q}_j \mathbf{x}_j \mathbf{x}_j^T \mathbf{Q}_j^T. \quad (59)$$



Following the same approach as (31), for  $n \rightarrow \infty$

$$\begin{aligned} E\{\lim_{n \rightarrow \infty} \tilde{\mathbf{R}}_n\} &= E\left\{\lim_{n \rightarrow \infty} (\lambda^{n-1} \mathbf{Q}_1 \mathbf{x}_1 \mathbf{x}_1^T \mathbf{Q}_1^T \right. \\ &\quad \left. + \dots + \mathbf{Q}_n \mathbf{x}_n \mathbf{x}_n^T \mathbf{Q}_n^T)\right\} \\ &= \lim_{n \rightarrow \infty} \frac{M\tilde{\sigma}_x^2}{L} [\lambda^{n-1} \mathbf{I} + \lambda^{n-2} \mathbf{I} + \dots + \mathbf{I}] \\ &= \frac{1}{1-\lambda} \frac{M\tilde{\sigma}_x^2}{L} \mathbf{I}, \end{aligned} \quad (60)$$

and hence it follows from (58)

$$\Gamma_n = (1-\lambda) \frac{L}{M\tilde{\sigma}_x^2} \mathbf{Q}_n. \quad (61)$$

Using (61), we may express the following terms in (18) as

$$\begin{aligned} E\{\Gamma_n \mathbf{x}_n \mathbf{x}_n^T\} &= E\{\mathbf{x}_n \mathbf{x}_n^T \Gamma_n^T\} \\ &= (1-\lambda) \frac{L}{M\tilde{\sigma}_x^2} E\{\mathbf{Q}_n \mathbf{x}_n \mathbf{x}_n^T\} \\ &= (1-\lambda) \mathbf{I} \\ E\{\Gamma_n \mathbf{x}_n \mathbf{x}_n^T \mathbf{v}_n \mathbf{v}_n^T \mathbf{x}_n \mathbf{x}_n^T \Gamma_n^T\} &= \left[\frac{(1-\lambda)L}{M\tilde{\sigma}_x^2}\right]^2 \times E\{\mathbf{Q}_n \mathbf{x}_n \mathbf{x}_n^T \mathbf{x}_n \mathbf{x}_n^T \mathbf{Q}_n\} \mathbf{R}_{\mathbf{v},n} \\ &= \frac{(1-\lambda)^2 L(L+2) \sigma_x^2}{M\tilde{\sigma}_x^2} \mathbf{R}_{\mathbf{v},n}, \\ E\{\Gamma_n \mathbf{x}_n \mathbf{x}_n^T \Gamma_n^T\} &= \left[\frac{(1-\lambda)L}{M\tilde{\sigma}_x^2}\right]^2 E\{\mathbf{Q}_n \mathbf{x}_n \mathbf{x}_n^T \mathbf{Q}_n\} \\ &= \frac{(1-\lambda)^2 L}{M\tilde{\sigma}_x^2} \mathbf{I}. \end{aligned} \quad (62)$$

Using the same approach as (47), by substituting the set of equations in (62) into (18) for  $K = 1$

$$\begin{aligned} \mathbf{R}_{\mathbf{v},n+1} &= \mathbf{R}_{\mathbf{v},n} - 2(1-\lambda) \mathbf{R}_{\mathbf{v},n} + \frac{(1-\lambda)^2 L(L+2) \sigma_x^2}{M\tilde{\sigma}_x^2} \mathbf{R}_{\mathbf{v},n} \\ &\quad + \frac{(1-\lambda)^2 L}{M\tilde{\sigma}_x^2} \sigma_w^2 \mathbf{I} + 2(1-\xi) \sigma_s^2 \mathbf{I}. \end{aligned} \quad (63)$$

As before, we assume that  $\mathbf{R}_{\mathbf{v},n}$  is fluctuating around its mean such that  $\mathbf{R}_{\mathbf{v}}$  is the approximately time-invariant autocorrelation matrix of the mean weight error vector. Defining  $\eta' = \text{tr}\{\mathbf{R}_{\mathbf{v}}\}$  we may then express (63) as

$$\begin{aligned} \eta'_{\text{MMax-RLS}} &= \eta' - 2(1-\lambda) \eta' + \frac{(1-\lambda)^2 L(L+2) \sigma_x^2}{M\tilde{\sigma}_x^2} \eta' \\ &\quad + \frac{(1-\lambda)^2 L^2 \sigma_w^2}{M\tilde{\sigma}_x^2} + 2L(1-\xi) \sigma_s^2 \\ &= \frac{(1-\lambda) L^2 \sigma_w^2}{\beta} + \frac{2L(1-\xi) \sigma_s^2 M\tilde{\sigma}_x^2}{(1-\lambda) \beta} \end{aligned} \quad (64)$$

where

$$\begin{aligned} \beta &= 2M\tilde{\sigma}_x^2 - (1-\lambda)L(L+2)\sigma_x^2 \\ &= L\sigma_x^2 [2\psi^{-1} - (1-\lambda)(L+2)] \end{aligned} \quad (65)$$

such that  $\psi$  is defined in (54). Comparing (64) and (33), we note that for  $M = L$  and  $\tilde{\sigma}_x^2 = \sigma_x^2$ ,  $\eta'_{\text{MMax-RLS}} = \eta'_{\text{RLS}}$  as expected. More importantly, the estimation variance for MMax-RLS is dependent on  $M$ . As can be seen from (64),  $\beta$  is a decreasing function of  $\psi$  and hence for a time-invariant system with  $\xi = 1$ , the steady-state misalignment  $\eta'_{\text{MMax-RLS}}$  is a decreasing function of  $M$ . This is contrary to the MMax-NLMS case as shown in (52) where the normalized misalignment is independent of  $M$  for time-invariant systems. Simulation results illustrating the dependency of the steady-state misalignment on  $M$  for MMax-RLS under the condition  $\xi = 1$  can be found in [12].

### C. Misalignment Analysis for MMax-AP

The tap-update equation for the MMax-AP algorithm can be written in a similar form to (12) where we have for  $j = 0, 1, \dots, K-1$

$$\Gamma_{n-j} = \mu_{n-j} \mathbf{Q}_{n-j} \quad (66)$$

such that  $\mu_{n-j} = 2\mu/(\mathbf{x}_{n-j}^T \mathbf{x}_{n-j})$ . Assuming that the input signal is drawn from an i.i.d process and statistically stationary, we may then evaluate

$$E\{\mu_{n-j}\} = c = \frac{2\mu}{L\sigma_x^2}. \quad (67)$$

We note again the condition  $E\{\Gamma_n\} = \Gamma$  is not valid here so we evaluate  $\text{tr}\{\mathbf{R}_{\mathbf{v},n}\}$  using the same approach as (46) giving

$$\begin{aligned} &\text{tr}\left\{K\sigma_w^2 \sum_{j=0}^{K-1} E\{\Gamma_{n-j} \mathbf{x}_{n-j} \mathbf{x}_{n-j}^T \Gamma_{n-j}^T\}\right\} \\ &= K\sigma_w^2 \text{tr}\left\{E\{\mu_{n-j}^2\} \sum_{j=0}^{K-1} E\{\mathbf{Q}_{n-j} \mathbf{x}_{n-j} \mathbf{x}_{n-j}^T \mathbf{Q}_{n-j}\}\right\} \\ &= K^2 \sigma_w c^2 M\tilde{\sigma}_x^2 \\ &\text{tr}\left\{\mathbf{R}_{\mathbf{v},n} \sum_{j=0}^{K-1} E\{\mathbf{x}_{n-j} \mathbf{x}_{n-j}^T \Gamma_{n-j}^T\}\right\} \\ &= \text{tr}\left\{\mathbf{R}_{\mathbf{v},n} \sum_{j=0}^{K-1} E\{\Gamma_{n-j}^T \mathbf{x}_{n-j} \mathbf{x}_{n-j}^T\}\right\} \\ &= \text{tr}\{\mathbf{R}_{\mathbf{v},n}\} K c \frac{M}{L} \tilde{\sigma}_x^2 \\ &\text{tr}\left\{E\left\{\sum_{j=0}^{K-1} \Gamma_{n-j} \mathbf{x}_{n-j} \mathbf{x}_{n-j}^T \mathbf{v}_{n-j} \mathbf{v}_{n-j}^T \right. \right. \\ &\quad \left. \left. \times \sum_{i=0}^{K-1} \mathbf{v}_{n-i}^T \mathbf{x}_{n-i} \mathbf{x}_{n-i}^T \Gamma_{n-i}^T \right\}\right\} \\ &= \text{tr}\{\mathbf{R}_{\mathbf{v},n}\} c^2 (L+2) \frac{M}{L} \tilde{\sigma}_x^2 \sigma_x^2 K^2. \end{aligned} \quad (68)$$

Substituting (68) into (18), we obtain

$$\begin{aligned} \text{tr}\{\mathbf{R}_{\mathbf{v},n+1}\} &= \text{tr}\{\mathbf{R}_{\mathbf{v},n}\} + 2(1-\xi) \sigma_s^2 L + K^2 \sigma_w^2 c^2 M\tilde{\sigma}_x^2 \\ &\quad - 2\text{tr}\{\mathbf{R}_{\mathbf{v},n}\} c \frac{M}{L} \tilde{\sigma}_x^2 K \\ &\quad + \text{tr}\{\mathbf{R}_{\mathbf{v},n}\} K^2 c^2 (L+2) \frac{M}{L} \tilde{\sigma}_x^2 \sigma_x^2 K^2. \end{aligned} \quad (69)$$

TABLE VI  
STEADY-STATE MISALIGNMENT FOR i.i.d. INPUT

Algorithm	$\eta'$
LMS	$\frac{\mu\sigma_w^2 L}{\phi} + \frac{(1-\xi)L\sigma_s^2}{2\mu\sigma_x^2\phi}$
NLMS	$\frac{\mu\sigma_w^2}{\sigma_x^2\phi} + \frac{(1-\xi)L^2\sigma_s^2}{2\mu\phi}$
AP	$\frac{K\mu\sigma_w^2}{\sigma_x^2\phi_a} + \frac{(1-\xi)\sigma_s^2 L^2}{2K\mu\phi_a}$
RLS	$\frac{(1-\lambda)L\sigma_w^2}{2\sigma_x^2\phi} + \frac{(1-\xi)L\sigma_s^2}{(1-\lambda)\phi}$
MMax-NLMS	$\frac{\mu\sigma_w^2}{\sigma_x^2\phi} + \frac{L}{M\tilde{\sigma}_x^2} \frac{(1-\xi)L^2\sigma_s^2\sigma_x^2}{2\mu\phi}$
MMax-AP	$\frac{K\mu\sigma_w^2}{\sigma_x^2\phi_a} + \frac{L}{M\tilde{\sigma}_x^2} \frac{(1-\xi)L^2\sigma_s^2\sigma_x^2}{2\mu K\phi_a}$
MMax-RLS	$\frac{(1-\lambda)L^2\sigma_w^2}{\beta} + \frac{(1-\xi)2L\sigma_s^2 M\tilde{\sigma}_x^2}{(1-\lambda)\beta}$

As before, we assume  $\mathbf{R}_{\mathbf{v},n}$  is fluctuating around its mean and substituting  $c = 2\mu/(L\sigma_x^2)$ , we can thus simplify (69) and write the misalignment as

$$\eta'_{\text{MMax-AP}} = \frac{K\mu\sigma_w^2}{\sigma_x^2\phi_a} + \frac{L\sigma_x^2}{M\tilde{\sigma}_x^2} \frac{(1-\xi)L^2\sigma_s^2}{2\mu K\phi_a} \quad (70)$$

where  $\phi_a$  is given in (41). For  $K = 1$ , the convergence performance of MMax-AP is the same as MMax-NLMS as expected. When  $M = L$ , we have  $\tilde{\sigma}_x^2 = \sigma_x^2$  and MMax-AP is equivalent to AP. We also note that, as for MMax-NLMS, the estimation variance is independent of  $M$  and so, as shown in [12], for  $\xi = 1$ , the same steady-state misalignment can be achieved for various  $M$ . In addition, the lag variance is proportional to  $\psi$  as defined in (54) and hence, degradation in misalignment performance is expected for reducing  $M$  when  $\xi < 1$ .

The steady-state misalignment of various algorithms for time-varying system conditions is summarized in Table VI.

## V. COMPUTATIONAL COMPLEXITY

MMax tap selection requires a sorting operation to select the  $M$  largest tap-inputs at each iteration. This can be achieved efficiently using, for example, the SORTLINE [30] or the short-sort [31] routine. A full description and examples of the short-sort approach are given in [31]. In summary, however, the short-sort algorithm selects  $A$  out of the first  $S < L$  elements of the tap input vector once every  $S$  sample periods. The coefficients corresponding to the  $A$  selected elements are updated and the algorithm then tracks the selected tap inputs as they propagate through the memory of the filter. The value of  $A$  is chosen such that in the SM-NLMS algorithm [31] for example,  $M$  out of  $L$  taps are updated per iteration on average and  $S \ll L$  typically. Thus, the worst case comparison load using short-sort is  $[1 + S - A]A/S$  comparisons per iteration compared to  $2 + 2\log_2 L$  used in the SORTLINE procedure.

Although many factors contribute to the complexity of an algorithm, in this paper we assess relative complexity in terms of the total number of additions, multiplications, and comparisons per sample period. The computational complexity for each algorithm is summarized in Tables VII and VIII while Table IX shows the computational complexity for an example case of  $L = 1024$ ,  $M = 512$ ,  $A = 64$ ,  $S = 128$ , and  $K = 2$ .

TABLE VII  
RELATIVE COMPLEXITY IN TERMS OF ADDITIONS AND MULTIPLICATIONS

Algorithm	Additions	Multiplications
SM-NLMS	$L + S + 2$ $+(A + 2)/S$	$L + S + 3$ $+(A + 3)/S$
MMax-NLMS	$L + M + 2$	$L + M + 3$
NLMS	$2L + 2$	$2L + 3$
MMax-AP	$L(K + 1) - K$ $+ M(K - 1) + 1$	$7K^2 + K(M$ $+ L) + L$
AP	$2LK - K + 1$	$7K^2 + 2KL + L$
MMax-RLS	$L^2 + L(3M - 1)$ $+ M - 1$	$2L^2 + 3L$ $\times (M + 1) + M$
RLS	$4L^2 - 1$	$5L^2 + 4L$

TABLE VIII  
RELATIVE COMPLEXITY IN TERMS OF COMPARISONS

Algorithm	Comparisons
SM-NLMS	$(S + 1 - A)A/S$
MMax-NLMS	$2\log_2 L + 2$
NLMS	0
MMax-AP	$2\log_2 L + 2$
AP	0
MMax-RLS	$2\log_2 L + 2$
RLS	0

TABLE IX  
AN EXAMPLE OF RELATIVE COMPLEXITY FOR  $[L = 1024$ ,  
 $M = 512, A = 64, S = 128, K = 2]$

Algorithm	Additions	Multiplications	Comparisons
SM-NLMS	1155	1156	33
MMax-NLMS	1538	1539	22
NLMS	2050	2051	0
MMax-AP	3583	4124	22
AP	4095	5148	0
MMax-RLS	$2.62 \times 10^6$	$3.67 \times 10^6$	22
RLS	$4.19 \times 10^6$	$5.25 \times 10^6$	0

The complexity of the MMax-AP algorithm is computed using the generalized Levinson algorithm [32]. As an illustrative example, an acoustic impulse response of 128 ms at 8-kHz sampling frequency corresponds to  $L = 1024$  for which the number of multiplications required by MMax-NLMS, MMax-AP, and MMax-RLS employing the SORTLINE algorithm with  $M = 512$  is approximately 75.0%, 80.1%, and 69.9% of the number for NLMS, AP, and RLS, respectively.

## VI. SIMULATIONS AND RESULTS

### A. Effect of Time Variation for NLMS and MMax-NLMS

We first present NLMS and MMax-NLMS simulations to support the analysis of normalized misalignment for time-varying system identification such as shown in Fig. 1. We employ the normalized misalignment  $\eta$  defined in (20). Fig. 4 shows NLMS results for a time-invariant system ( $\xi = 1$ ) and three time-varying systems ( $\xi = \{0.999999, 0.99999, 0.9999\}$ ) where smaller values of  $\xi$  indicate higher degrees of time-variation. The adaptive filter is of length  $L = 64$  while the adaptive step-size is  $\mu = 0.1$  in these examples. This allows the NLMS algorithm to track the unknown system. The learning curves are averaged over five independent trials, and the theoretical values of  $\eta'_{\text{NLMS}}$  given by (26) are superimposed as straight horizontal lines. Fig. 5 shows the results of an equivalent experiment for

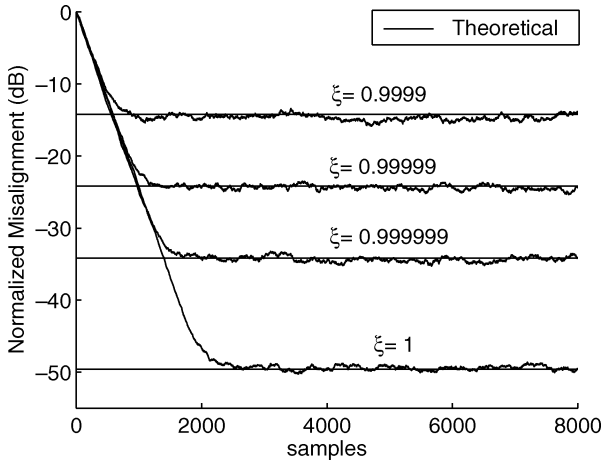


Fig. 4. NLMS normalized misalignment for varying  $\xi$  with  $L = 64$ ,  $\mu = 0.1$ ,  $\sigma_x^2 = \sigma_s^2 = 1$ , SNR = 40 dB.

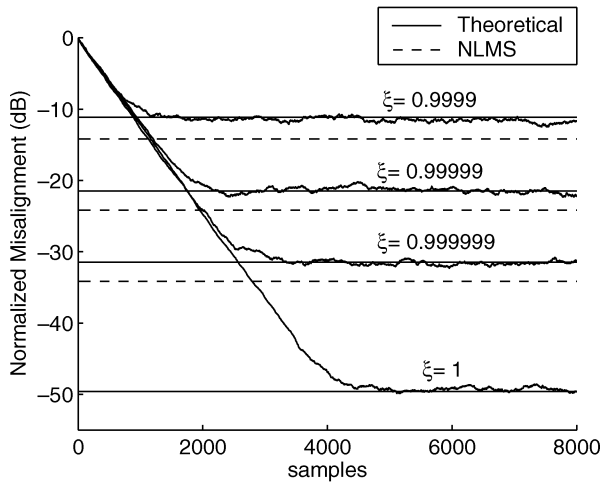


Fig. 5. MMax-NLMS normalized misalignment for varying  $\xi$  with  $\mu = 0.1$ ,  $\sigma_x^2 = \sigma_s^2 = 1$ ,  $L = 64$ ,  $M = 8$ , SNR = 40 dB. Dashed lines indicate corresponding performance for NLMS.

MMax-NLMS with  $L = 64$  and  $M = 8$ . For comparison purposes, the corresponding theoretical values of  $\eta_{\text{NLMS}}$  from the previous experiment are also included in Fig. 5 as dashed lines. For both experiments, white Gaussian measurement noise  $w_n$  is added such that the SNR of 40 dB is achieved.

The results show that both NLMS and MMax-NLMS are sensitive to time variation of the unknown system in that the misalignment performance degrades with increasing deviation of  $\xi$  from unity. The MMax-NLMS algorithm for example can be seen to perform around 3 to 4 dB worse, in terms of steady-state normalized misalignment, than NLMS under these time-varying conditions. For a time-invariant system,  $\xi = 1$ , both MMax-NLMS and NLMS achieve the same steady-state misalignment since their estimation variance are equivalent as can be seen by (26) and (52). The MMax-NLMS algorithm, however, has a lower rate of convergence compared to that of NLMS as expected irrespective of time-variation.

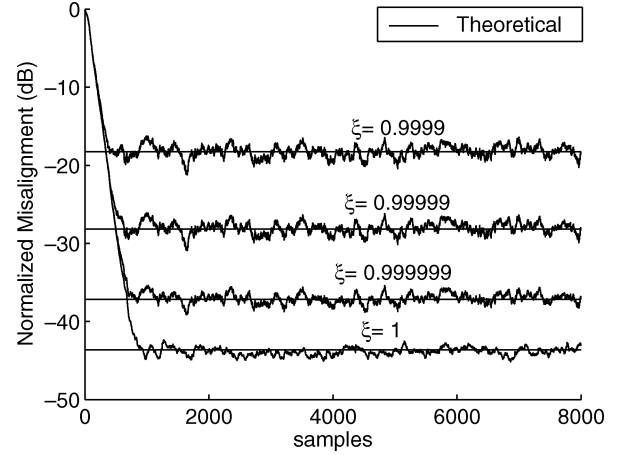


Fig. 6. AP normalized misalignment for varying  $\xi$  with  $\mu = 0.1$ ,  $\sigma_x^2 = \sigma_s^2 = 1$ ,  $L = 64$ ,  $K = 3$ , SNR = 40 dB.

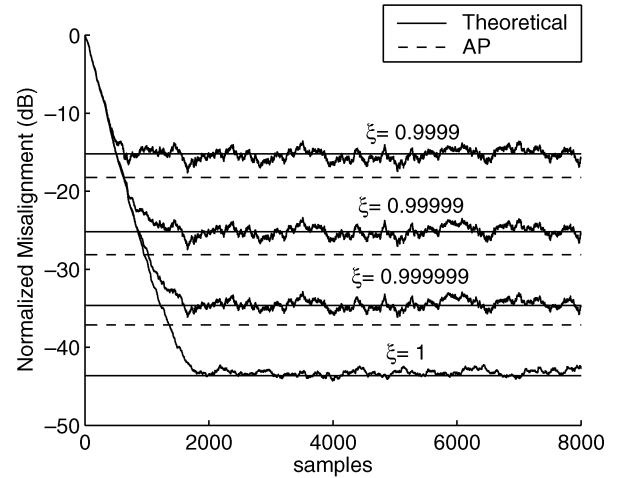


Fig. 7. MMax-AP normalized misalignment for varying  $\xi$  with  $\mu = 0.1$ ,  $\sigma_x^2 = \sigma_s^2 = 1$ ,  $L = 64$ ,  $M = 8$ ,  $K = 3$ , SNR = 40 dB. Dashed lines indicate corresponding performance for AP.

### B. Effect of Time-Variation for AP and MMax-AP

Figs. 6 and 7 show the normalized misalignment for the AP and MMax-AP algorithms, respectively, where the straight lines indicate theoretical misalignment for various time-varying conditions  $\xi$ . The dashed lines in Fig. 7 represent the normalized misalignment for the fully updated AP algorithm. In these simulations, the adaptive filter is of length  $L = 64$  with  $\mu = 0.1$ . We have used projection order  $K = 3$  as an illustration while  $w_n$  is added to achieve an SNR of 40 dB. For the MMax-AP algorithm,  $M = 8$  was used. In both simulations, the learning curves are averaged over five independent trials.

The results indicate that the normalized misalignment performance degrades for both AP and MMax-AP by approximately 2 to 3 dB with increasing deviation of  $\xi$  from unity. For a time-invariant system,  $\xi = 1$ , the normalized steady-state misalignment of the MMax-AP is insensitive to the tap selection as can be seen in the estimation variance term in (70).

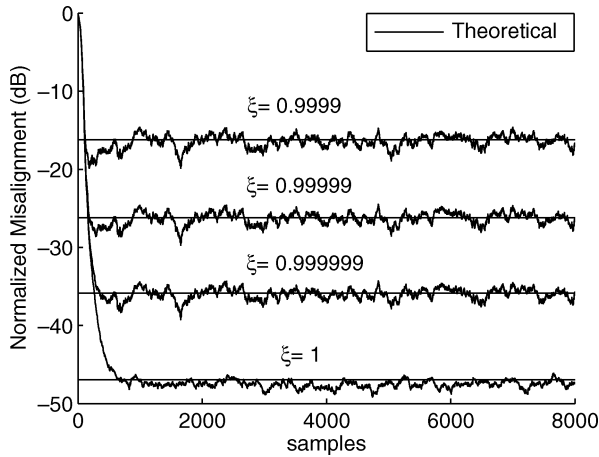


Fig. 8. RLS normalized misalignment for varying  $\xi$  with  $\lambda = 0.9948$ ,  $\sigma_x^2 = \sigma_s^2 = 1$ ,  $L = 64$ , SNR = 40 dB.

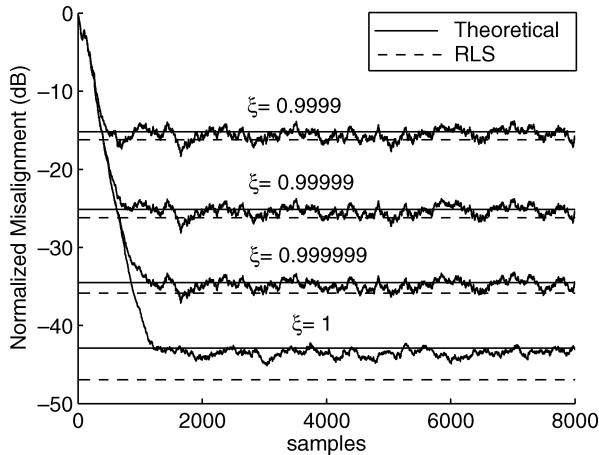


Fig. 9. MMax-RLS normalized misalignment for varying  $\xi$  with  $\lambda = 0.9948$ ,  $\sigma_x^2 = \sigma_s^2 = 1$ ,  $L = 64$ ,  $M = 8$ , SNR = 40 dB. Dashed lines indicate corresponding performance for RLS.

### C. Effect of Time Variation for RLS and MMax-RLS

Figs. 8 and 9 show RLS and MMax-RLS normalized misalignment results for various  $\xi$  as before. In these simulations, the adaptive filter is of length  $L = 64$  and a forgetting factor of  $\lambda = 1 - 1/(3L) = 0.9948$  is used [33]. With reference to Fig. 1 and (14),  $\sigma_s^2 = \sigma_x^2 = 1$  while  $w_n$  is added to achieve an SNR of 40 dB for each of the five independent trials. For the MMax-RLS algorithm,  $M = 8$  taps were selected for adaptation at each iteration.

The results show that both RLS and MMax-RLS are sensitive to time-variation of the unknown system with misalignment performance degrading by approximately 7 and 3 dB for  $\xi = 1$  and  $\xi = 0.9999$ , respectively. Unlike the MMax-NLMS and MMax-AP algorithms, MMax-RLS is sensitive to tap selection even for an unknown system with  $\xi = 1$  as can be seen by the term  $\beta$  in the estimation variance of (64).

### D. Effect of Step-Size for NLMS and MMax-NLMS

Fig. 10 shows the effect of different values of  $\mu$  on the steady-state misalignment for NLMS under stationary ( $\xi = 1$ ) and time-varying ( $\xi = 0.99999$ ) cases. In this experiment, the

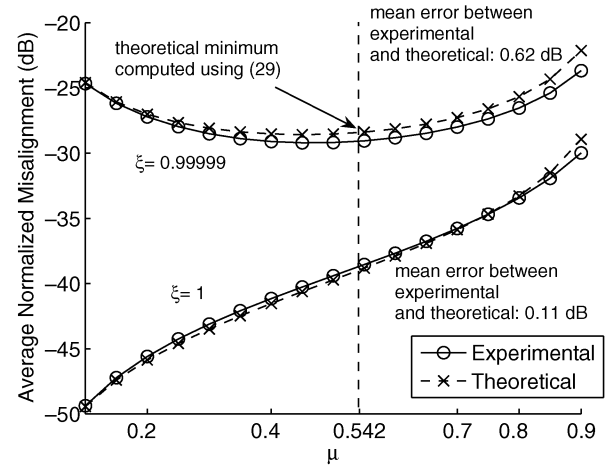


Fig. 10. NLMS: variation of average normalized misalignment with  $\mu$  for  $\sigma_x^2 = \sigma_s^2 = 1$ ,  $L = 64$ , SNR = 40 dB.

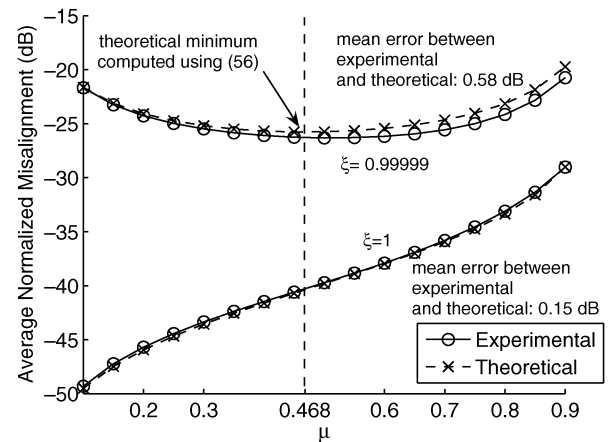


Fig. 11. MMax-NLMS: variation of average normalized misalignment with  $\mu$  for  $\sigma_x^2 = \sigma_s^2 = 1$ ,  $L = 64$ ,  $M = 8$ , SNR = 40 dB.

filter length was  $L = 64$ , and  $w_n$  was added at an SNR of 40 dB. The average steady-state normalized misalignment was obtained from five independent trials.

We observe that for the stationary case  $\xi = 1$ , the steady-state normalized misalignment increases with  $\mu$  as expected. In this simulation example, the mean difference between the experimental and theoretical steady-state normalized misalignment is 0.11 dB. For the case of  $\xi = 0.99999$ , we note that there exists a  $\mu_{\text{mis}}$  such that the lowest misalignment can be achieved. The theoretical value of  $\mu_{\text{mis}} = 0.542$ , computed using (29), is shown by the vertical dotted line. The mean difference between the experimental and the theoretical normalized misalignment is 0.62 dB.

Fig. 11 shows the effect of step-size on MMax-NLMS under the conditions  $\xi = 1$  and  $\xi = 0.99999$  with  $L = 64$  and  $M = 8$ . As before, we have simulated this experiment using 40-dB SNR. Similar to the case of NLMS, we observe that for  $\xi = 1$ , the steady-state normalized misalignment is approximately linear in  $\mu$ . For the case of  $\xi = 0.99999$ , there exists a  $\mu_{\text{mis}} = 0.468$  governed by (56) which is plotted as a vertical line. The mean difference between the experimental and theoretical steady-state normalized misalignment for the case of  $\xi = 1$  and  $\xi = 0.99999$

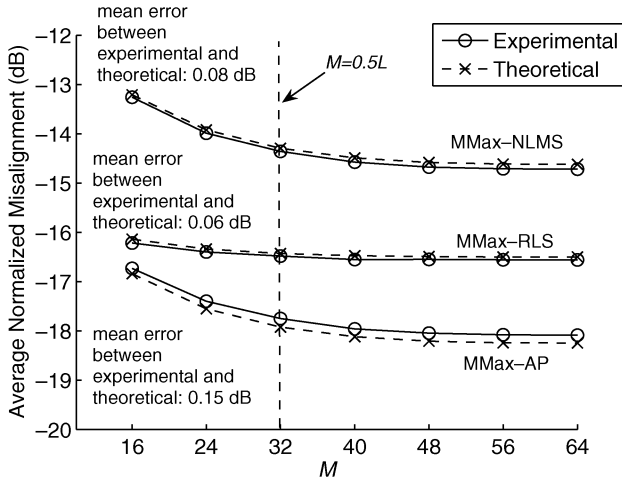


Fig. 12. Variation of average normalized misalignment with  $M$  for  $L = 64$ ,  $\mu = 0.1$ ,  $\lambda = 0.9948$ ,  $\xi = 0.9999$ ,  $\sigma_x^2 = \sigma_s^2 = 1$ , SNR = 40 dB.

are 0.15 and 0.58 dB, respectively. Comparing Figs. 10 and 11, we note that the normalized misalignment for MMax-NLMS is degraded for the time-varying unknown system condition compared to NLMS.

#### E. Effect of Tap Selection on Normalized Misalignment

We now compare the effect of tap selection on the normalized misalignment with a time-varying unknown system for the MMax-based algorithms. Fig. 12 shows the variation of average normalized misalignment with  $M$  for the case of MMax-NLMS, MMax-AP, and MMax-RLS. The length of the adaptive filter was  $L = 64$  while  $16 \leq M \leq 64$  and  $\xi = 0.9999$ . For the MMax-NLMS and MMax-AP algorithms  $\mu = 0.1$  while for MMax-RLS  $\lambda = 1 - 1/(3L) = 0.9948$  are used. The steady-state normalized misalignment for each algorithm is averaged over five independent trials and, for each trial, SNR = 40 dB.

Although the performance for each algorithm is plotted on the same axis, our intention here is not to compare each algorithm's relative normalized misalignment in this simulation example. We see instead that, for each algorithm, the normalized misalignment reduces with increasing  $M$  under the same time-varying condition and more importantly, for each algorithm, there is only a modest degradation in steady-state misalignment performance with reducing  $M$  for  $0.5L \leq M < L$ . When  $M$  is reduced further, the degradation in steady-state misalignment performance is more pronounced for the MMax-NLMS and MMax-AP algorithms since, as shown and discussed in Fig. 3,  $\psi$  is increased significantly. The degradation in steady-state normalized misalignment performance for MMax-RLS is less pronounced, suffering approximately 0.36-dB degradation from  $M = 64$  to  $M = 16$ . The mean error between theoretical and experimental results in this simulation for MMax-NLMS, MMax-AP, and MMax-RLS are 0.08, 0.15, and 0.06 dB, respectively.

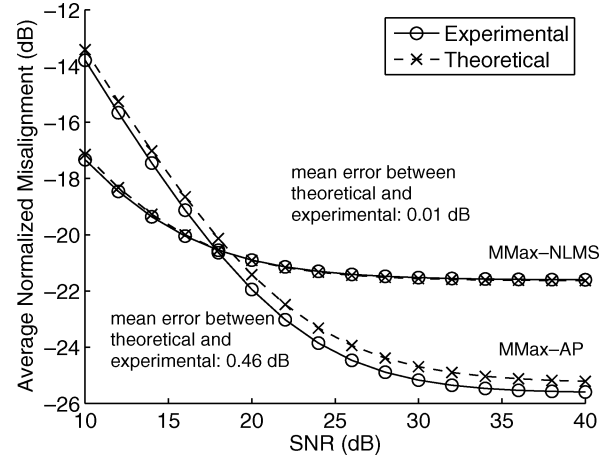


Fig. 13. MMax-NLMS and MMax-AP: variation of average normalized misalignment with SNR for  $L = 64$ ,  $M = 8$ ,  $\mu = 0.1$ ,  $\xi = 0.99999$ ,  $\sigma_x^2 = \sigma_s^2 = 1$ .

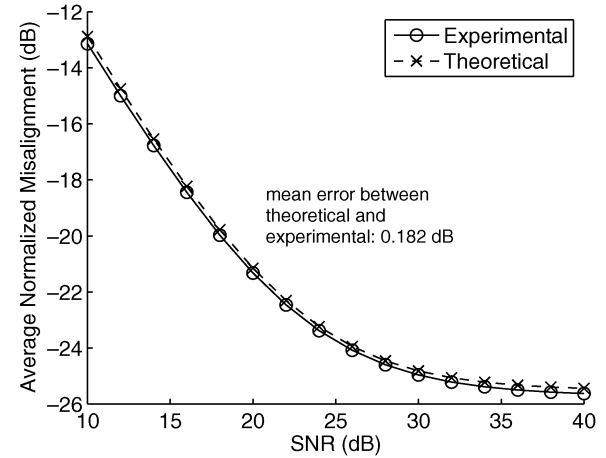


Fig. 14. MMax-RLS: variation of average normalized misalignment with SNR for  $L = 64$ ,  $M = 8$ ,  $\lambda = 0.9948$ ,  $\xi = 0.99999$ ,  $\sigma_x^2 = \sigma_s^2 = 1$ .

#### F. Effect of SNR on Normalized Misalignment

We now investigate the effect of SNR on normalized misalignment for the various MMax selective-tap algorithms under time-varying unknown system conditions. The experimental parameters for this simulation setup were  $L = 64$ ,  $M = 8$ ,  $\xi = 0.99999$ ,  $\mu = 0.1$ ,  $K=3$ , and  $\lambda = 1 - 1/(3L) = 0.9948$ . The normalized misalignment for each algorithm was averaged over five independent trials.

Fig. 13 shows the variation of MMax-NLMS and MMax-AP normalized misalignment with SNR. For each of the algorithms, we note that the normalized misalignment improves with increasing SNR as expected. The MMax-AP algorithm degrades more severely at low SNR and achieves an improvement of approximately 12-dB normalized misalignment when SNR is increased from 10 to 40 dB. On the contrary, the MMax-NLMS shows smoother degradation in misalignment for low SNR compared to that of MMax-AP.

Fig. 14 shows the corresponding normalized misalignment for MMax-RLS under various SNR conditions. As the SNR is

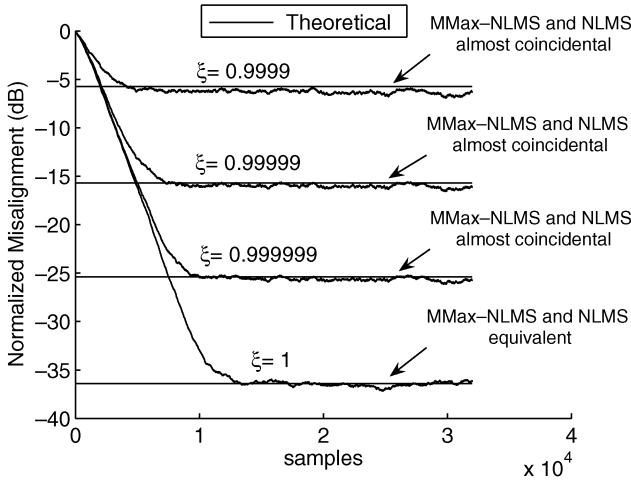


Fig. 15. Theoretical normalized misalignment for NLMS (dashed) and MMax-NLMS (solid) [ $L = 1024$ ,  $M = 128$ ,  $\mu = 0.7$ ,  $\text{SNR} = 40$  dB].

increased from 10 to 40 dB, we note that there is an improvement, though not linearly, of approximately 12.5 dB in normalized misalignment. The mean errors between our theoretical and experimental results are 0.01, 0.46, and 0.182 dB for MMax-NLMS, MMax-AP, and MMax-RLS, respectively.

### G. Illustrative Case With Large $L$

We present an example to illustrate the validity of the results obtained for the case when  $L$  is large, as for acoustic echo cancellation where an adaptive filter length of  $L = 1024$  taps may be required. Fig. 15 shows results for the MMax-NLMS with  $M = 128$  and  $\mu = 0.7$  and  $\text{SNR} = 40$  dB. As before, the theoretical steady-state normalized misalignment for the MMax-NLMS algorithm is plotted as straight horizontal solid lines. The learning curves for the MMax-NLMS are averaged over five independent trials. The results show that the steady-state misalignment for the MMax-NLMS is coincidental to NLMS for time-varying system conditions. As before, for  $\xi = 1$ ,  $\eta_{\text{NLMS}} = \eta_{\text{MMax-NLMS}}$ .

## VII. DISCUSSION AND CONCLUSION

We have presented a class of MMax selective-tap adaptive filters for system identification. A misalignment analysis for these algorithms was presented which describes the performance of adaptive filters when tracking a time-varying unknown system that varies according to a modified first-order Markov model [19]. This analysis can be applied to all algorithms that can be written using the update equation given in (12). When the time-variation is removed by setting  $\xi = 1$ , the analysis yields the known results [1] for the various fully updated algorithms. The results for the standard algorithms are consistent with previous work [1], [18] in terms of estimation variance but offer new insights for the lag variance. Under time-invariant system conditions, the steady-state normalized misalignment for MMax-NLMS and MMax-AP is independent of  $M$  while the same is not true for MMax-RLS. For time-varying systems, the performance of MMax-based algorithms in terms of steady-state misalignment degrades with increasing time-variation. This degra-

dation is proportional to  $\psi$ , and as can be seen from Fig. 3, for  $0.5L \leq M < L$ ,  $\psi$  increases smoothly and consequently the degradation in steady-state performance is negligible. This property has been exploited in the exclusive-maximum (XM) tap selection, which has been deployed in SAEC algorithms to give good convergence performance such as presented in [13] and [14]. In addition, we have shown that under time-varying unknown system conditions, there exists for NLMS and MMax-NLMS, an optimal step-size given by (29) and (56), respectively, which jointly maximizes the performances in terms of low misalignment and high convergence rate. Simulations verify that the analysis accurately describes the performance of the algorithms. This analysis enables a judicious tradeoff between the computational savings of selective-tap algorithms and their tracking performance.

## APPENDIX I

### FOURTH-ORDER FACTORIZATION FOR ZERO MEAN GAUSSIAN VARIABLES

For an i.i.d. Gaussian distributed signal  $x(n)$ , the matrix  $\Psi = E[\mathbf{x}_n \mathbf{x}_n^T \mathbf{x}_n \mathbf{x}_n^T]$  has elements

$$\Psi_{k,l} = E \left[ x(n-k) \sum_{i=1}^L x^2(n-i) x(n-l) \right]$$

where  $\mathbf{x}_n = [x(n), x(n-1), \dots, x(n-L+1)]^T$ . The factorization property of real zero-mean Gaussian variables is that

$$E[x(i)x(j)x(k)x(l)] = E[x(i)x(j)]E[x(k)x(l)] + E[x(i)x(k)]E[x(j)x(l)] + E[x(i)x(l)]E[x(j)x(k)]$$

from which

$$E[\mathbf{x}_n \mathbf{x}_n^T \mathbf{x}_n \mathbf{x}_n^T]_{kl} = 2 \sum_{i=1}^L E[x(n-k)x(n-i)] \times E[x(n-l)x(n-i)] + E[x(n-k)x(n-l)] \sum_{i=1}^L E[x^2(n-i)].$$

From the above it can be seen that, for the complete matrix,  $\Psi = 2\mathbf{R}^2 + \mathbf{R}\text{tr}\{\mathbf{R}\}$ . Now for  $x(n)$  i.i.d Gaussian variables

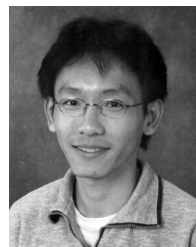
$$E[x(n-i)x(n-j)] = \begin{cases} 0, & i \neq j \\ \sigma_x^2, & i = j \end{cases}$$

so that  $\Psi = (L+2)\sigma_x^4 \mathbf{I}$ .

## REFERENCES

- [1] S. Haykin, *Adaptive Filter Theory*, ser. Information and System Science, 4th ed. Upper Saddle River, NJ: Prentice-Hall, 2002.
- [2] S. C. Douglas, "Adaptive filters employing partial updates," *IEEE Trans. Circuits Syst. II*, vol. 44, no. 3, pp. 209–216, Mar. 1997.
- [3] T. Aboulnasr and K. Mayyas, "MSE analysis of the M-Max NLMS adaptive algorithm," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, 1998, vol. 3, pp. 1669–1672.
- [4] T. Aboulnasr and K. Mayyas, "Complexity reduction of the NLMS algorithm via selective coefficient update," *IEEE Trans. Signal Process.*, vol. 47, no. 5, pp. 1421–1424, May 1999.
- [5] K. Dogancay and O. Tanrikulu, "Adaptive filtering algorithms with selective partial updates," *IEEE Trans. Circuits Syst. II*, vol. 48, no. 8, pp. 762–769, Aug. 2001.

- [6] S. L. Gay and S. Tavathia, "The fast affine projection algorithm," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, 1995, vol. 5, pp. 3023–3026.
- [7] D. L. Duttweiler, "Proportionate normalized least mean square adaptation in echo cancellers," *IEEE Trans. Speech Audio Process.*, vol. 8, no. 5, pp. 508–518, Sep. 2000.
- [8] J. Benesty and S. L. Gay, "An improved PNLMS algorithm," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, 2002, vol. 2, pp. 1881–1884.
- [9] A. Sugiyama, S. Ikeda, and A. Hirano, "A fast convergence algorithm for sparse-tap adaptive FIR filters identifying an unknown number of dispersive regions," *IEEE Trans. Signal Process.*, vol. 50, no. 12, pp. 3008–3017, Dec. 2002.
- [10] H. Deng and M. Doroslovacki, "Improving convergence of the PNLMS algorithm for sparse impulse response identification," *IEEE Signal Process. Lett.*, vol. 12, no. 3, pp. 181–184, Mar. 2005.
- [11] A. W. H. Khong and P. A. Naylor, "Selective-tap adaptive algorithms in the solution of the non-uniqueness problem for stereophonic acoustic echo cancellation," *IEEE Signal Process. Lett.*, vol. 12, no. 4, pp. 269–272, Apr. 2005.
- [12] P. A. Naylor and A. W. H. Khong, "Affine projection and recursive least squares adaptive filters employing partial updates," in *Proc. 38th Asilomar Conf. Signals, Syst., Comput.*, Nov. 2004, vol. 1, pp. 950–954.
- [13] A. W. H. Khong and P. A. Naylor, "A family of selective-tap algorithms for stereo acoustic echo cancellation," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, Mar. 2005, pp. 133–136.
- [14] A. W. H. Khong and P. A. Naylor, "Stereophonic acoustic echo cancellation employing selective-tap adaptive algorithms," *IEEE Trans. Audio, Speech, Lang. Process.*, vol. 14, no. 3, pp. 785–796, May 2006.
- [15] B. Widrow, J. M. McCool, M. G. Larimore, and C. R. Johnson, "Stationary and non-stationary learning characteristics of the LMS adaptive filter," *Proc. IEEE*, vol. 64, no. 8, pp. 1151–1162, Aug. 1976.
- [16] B. Widrow and E. Walach, "On the statistical efficiency of the LMS algorithm with nonstationary inputs," *IEEE Trans. Inf. Theory*, vol. IT-30, no. 2, pp. 211–221, Mar. 1984.
- [17] E. Eleftheriou and D. D. Falconer, "Tracking properties and steady-state performance of RLS adaptive algorithms," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. ASSP-34, no. 5, pp. 1087–1110, Sep. 1986.
- [18] O. Macchi, "Optimization of adaptive identification for time-varying filters," *IEEE Trans. Automat. Contr.*, vol. 31, no. 3, pp. 283–287, 1986.
- [19] N. J. Bershad, S. McLaughlin, and C. F. N. Cowan, "Performance comparison of RLS and LMS algorithms for tracking a first order Markov communications channel," in *Proc. IEEE Int. Symp. Circuits Syst.*, 1990, vol. 1, pp. 266–270.
- [20] W. Sherliker, "Acoustic echo cancellation algorithms with tap selection for non-stationary environments," Ph.D. dissertation, Imperial College London, London, U.K., 2000.
- [21] S. C. Douglas, "Analysis and implementation of the MAX-NLMS adaptive filter," in *Proc. 29th Asilomar Conf. Signals, Syst. Comput.*, 1995, vol. 1, pp. 659–663.
- [22] T. Aboulnasr and K. Mayyas, "Selective coefficient update of gradient-based adaptive algorithms," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, 1997, vol. 3, pp. 1929–1932.
- [23] D. J. Tylavsky and G. R. Sohie, "Generalization of the matrix inversion lemma," *Proc. IEEE*, vol. 74, no. 7, pp. 1050–1052, Jul. 1986.
- [24] V. Myllyla and G. Schmidt, "Pseudo-optimal regularization for affine projection," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, 2002, vol. 2, pp. 1917–1920.
- [25] S. G. Sankaran and A. A. Beex, "Convergence behaviour of affine projection algorithms," *IEEE Trans. Signal Process.*, vol. 48, no. 4, pp. 1086–1096, Apr. 2000.
- [26] J. B. Allen and D. A. Berkley, "Image method for efficiently simulating small-room acoustics," *J. Acoust. Soc. Amer.*, vol. 65, no. 4, pp. 943–950, Apr. 1979.
- [27] C. Breining, "Control of a hands-free telephone set," *Signal Process.*, vol. 61, pp. 131–143, 1997.
- [28] M. Godavarti and A. O. Hero, "Stability bounds on step-size for the partial update LMS algorithm," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, 1999, vol. 3, pp. 1677–1680.
- [29] M. Rupp, "Contraction mapping: An important property in adaptive filters," in *Proc. 6th IEEE Digital Signal Process. Workshop*, 1994, pp. 273–276.
- [30] I. Pitas, "Fast algorithms for running ordering and max/min calculation," *IEEE Trans. Circuits Syst.*, vol. 36, no. 6, pp. 795–804, Jun. 1989.
- [31] P. A. Naylor and W. Sherliker, "A short-sort M-max NLMS partial update adaptive filter with applications to echo cancellation," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, 2003, vol. 5, pp. 373–376.
- [32] S. L. Gay and J. Benesty, *Acoustic Signal Processing for Telecommunication*. Norwell, MA: Kluwer, 2001.
- [33] C. Breining, P. Dreiseitel, E. Hänsler, A. Mader, B. Nitsch, H. Puder, T. Schertler, G. Schmidt, and J. Tilp, "Acoustic echo control: An application of very-high-order adaptive filter," *IEEE Signal Process. Mag.*, vol. 16, no. 4, pp. 42–69, Jul. 1999.



**Andy W. H. Khong** (M'06) received the B.Eng. degree in electrical and electronic engineering from the Nanyang Technological University, Singapore, in 2002 and the Ph.D. degree from Imperial College London, London, U.K., in 2006.

Since 2006, he has been a Research Associate at Imperial College London. His research interests are mainly in the area of (blind and nonblind) adaptive algorithms for single and multichannel acoustic system identification with applications to echo control and speech enhancement. His industrial research experience includes source localization and classification using acoustic and seismic signals.



**Patrick A. Naylor** (SM'07) received the B.Eng. degree in electronics and electrical engineering from the University of Sheffield, Sheffield, U.K., in 1986 and the Ph.D. degree from Imperial College London, London, U.K., in 1990.

Since 1989, he has been a Member of Academic Staff in the Communications and Signal Processing Research Group, Imperial College London, where he is also Director of Postgraduate Studies. His research interests are in the areas of speech and audio signal processing, and he has worked in particular on adaptive signal processing for acoustic echo control, speaker identification, multi-channel speech enhancement, and speech production modeling. In addition to his academic research, he enjoys several fruitful links with industry in the U.K., USA, and in mainland Europe.

Dr. Naylor is an Associate Editor of the IEEE SIGNAL PROCESSING LETTERS and a member of the IEEE Signal Processing Society Technical Committee on Audio and Electroacoustics.