New Sequential Partial-Update Least Mean M-Estimate Algorithms for Robust Adaptive System Identification in Impulsive Noise

Y. Zhou, *Member, IEEE*, S. C. Chan, *Member, IEEE*, and K. L. Ho, *Senior Member, IEEE*

***Abstract*—The sequential partial-update least mean square (S-LMS)-based algorithms are efficient methods for reducing the arithmetic complexity in adaptive system identification and other industrial informatics applications. They are also attractive in acoustic applications where long impulse responses are encountered. A limitation of these algorithms is their degraded performances in an impulsive noise environment. This paper proposes new robust counterparts for the S-LMS family based on M-estimation. The proposed sequential least mean M-estimate (S-LMM) family of algorithms employ nonlinearity to improve their robustness to impulsive noise. Another contribution of this paper is the presentation of a convergence performance analysis for the S-LMS/S-LMM family for Gaussian inputs and additive Gaussian or contaminated Gaussian noises. The analysis is important for engineers to understand the behaviors of these algorithms and to select appropriate parameters for practical realizations. The theoretical analyses reveal the advantages of input normalization and the M-estimation in combating impulsive noise. Computer simulations on system identification and joint active noise and acoustic echo cancellations in automobiles with double-talk are conducted to verify the theoretical results and the effectiveness of the proposed algorithms.**

***Index Terms*—Adaptive echo cancellation (AEC), adaptive noise cancellation (ANC), double-talk, impulsive noise, least mean M-estimate (LMM), least mean square (LMS), partial-update adaptive filters, performance analysis, system identification.**

I. Introduction

SYSTEM measurement/identification is frequently encountered in many applications such as adaptive control, acoustic and speech processing, digital communications [1], ultrasonic nondestructive testing [2], power system measurement [3], interference suppression in industrial and biomedical engineering [4], etc. To cater for time-varying signal statistics and to reduce arithmetic complexity, adaptive filtering algorithms are usually employed to iteratively identify the impulse response of unknown linear systems [5]. The well-known least mean square (LMS) [6] algorithm and its input normalization variant, the normalized LMS (NLMS) algorithm [7], are widely used in many system identification problems because of their numerical stability and computational simplicity. However, in some applications such as acoustic signal processing and network echo cancellation, higher order adaptive filters are usually required to model the acoustic paths with long impulse responses. Many algorithms have been proposed to reduce the power consumption and arithmetic and implementation complexities of the LMS and NLMS algorithms. Partial update (PU) [8]–[13] is an efficient technique which only updates a portion of the filter coefficients at each iteration. It is very attractive for implementation in hardware, very large scale integration, and digital signal processors.

PU adaptive filtering algorithms can be broadly categorized into two classes. The first class updates the coefficients using certain data-dependent selection criteria. Examples are the M-max NLMS algorithm [9], the selective-block-update NLMS algorithm [10], the L_∞-norm-based algorithm [11], the set-membership PU-NLMS algorithm [12], etc. These algorithms may have a faster convergence rate than those using fixed updating strategies. However, coefficient selection unavoidably increases the computational complexity and may suffer from convergence problems for nonstationary signals due to data-dependent updating [13]. On the other hand, the second class of algorithms uses predetermined updating schemes to update the filter coefficients. Representative algorithms include the periodic LMS algorithm [14] which updates all the filter coefficients periodically at every Pth iteration and the sequential PU LMS (S-LMS) algorithm [14] which partitions the coefficients into nonoverlapping groups and updates them sequentially at each iteration. These algorithms are simpler to implement and are found to be more stable for certain nonstationary signals than the first class of algorithms [13]. Furthermore, a new stochastic PU LMS (SPU-LMS) algorithm, which randomly schedules coefficient updating, was proposed in [13] to improve the stability of the S-LMS algorithm. Other variants of the S-LMS algorithm include the sequential block LMS (SB-LMS) algorithm for active noise control problems [15] and the sequential block NLMS (SB-NLMS) algorithm [16], which are the block implementation of the S-LMS algorithm and its normalized version, the sequential NLMS (S-NLMS) algorithm, respectively. The latter has also been used in adaptive echo cancellation (AEC) systems to reduce the implementation complexity [17]. Another S-NLMS algorithm with modified Huber (MH) nonlinearity, called the sequential

Manuscript received March 18, 2010; revised May 28, 2010 and October 5, 2010; accepted November 11, 2010. Date of publication December 10, 2010; date of current version August 12, 2011.

Y. Zhou is with the Key Laboratory of Noise and Vibration Research, Institute of Acoustics, Chinese Academy of Sciences, Beijing 100080, China (e-mail: yizhou@mail.ioa.ac.cn).

S. C. Chan and K. L. Ho are with the Department of Electrical and Electronic Engineering, The University of Hong Kong, Pokfulam, Hong Kong (e-mail: scchan@eee.hku.hk; klho@eee.hku.hk).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TIE.2010.2098359

0278-0046/$26.00 © 2010 IEEE

block PU normalized least mean M-estimate (LMM) algorithm, was also proposed and analyzed in [18], where a different form of normalization based on the constrained minimization approach in [10] was developed. Because of the difference in the normalization used, its convergence analysis is also considerably different from those studied in this paper. In this paper, we collectively call the S-LMS/NLMS, the SB-LMS/NLMS, and the SPU-LMS/NLMS algorithms as the S-LMS family of algorithms.

Driven by the practical implementation advantages of the S-LMS family, there is also considerable interest in their performance analysis. Important performance measures for adaptive filters include the initial convergence rate, the steady-state excess mean square error (MSE) (EMSE) over the ideal Wiener filter after convergence, the ability to track time-varying systems, and the computational complexity. Other important design issues include the maximum step size to achieve stable operation and to achieve a desired EMSE. Results concerning the convergence behavior of the S-LMS algorithm under the stationary signal and small step size assumptions were reported in [14]. Similar conclusions and convergence conditions for the SPU-LMS algorithm have also been obtained recently in [13].

A disadvantage of the S-LMS family of algorithms is that they are based on least square (LS) estimation as in the conventional LMS algorithm, and hence, their performance will deteriorate considerably when the desired signal is corrupted by impulsive noise [20], [24]–[28], [32]–[34], [37], [44], [45]. Such noise may arise naturally or from other man-made interference. In this paper, we derived a new class of robust S-LMS algorithms, called the sequential LMM (S-LMM) family, with improved performance in an impulsive noise environment. They are nonlinear versions of the conventional S-LMS family and are based on robust M-estimation [19] and adaptive threshold selection (ATS). The latter two techniques have been successfully employed in the LMM and transform domain normalized LMM algorithms [20] for robust filtering in an impulsive noise environment. The resultant algorithms are called the S-LMM/NLMM, SB-LMM/NLMM, and the SPU-LMM/NLMM algorithms, respectively. They are also collectively called the S-LMM family of algorithms.

Another contribution of this paper is a study of the convergence performance of the S-LMS and S-LMM families of algorithms. This is particularly useful to speech processing and other applications, where Gaussian random processes are commonly employed to model the signals involved. The mean and mean square convergence analyses of these algorithms are treated in a single framework using Price's theorem [21], [22] with Gaussian inputs and additive Gaussian or contaminated Gaussian (CG) noises. The novelty of the analysis lies in handling the normalization, evaluating the expectations that are specific to the S-LMS family of algorithms, and dealing with the error nonlinearity. To our best knowledge, the performance analysis of the S-NLMS, the SB-LMS/NLMS, and the SPU-NLMS algorithms and their nonlinear versions are unavailable in the literature. The results so obtained comply with the conclusions of previous works in [13] and [14] for the PU-LMS algorithm and also provide new findings due to the effects of normalization and nonlinearity. Moreover, when the decimation factor C is equal to one, all the analyses will reduce to those of the standard LMS/LMM/NLMS/NLMM algorithms. The robustness of the new S-LMM family of algorithms in CG noise is also theoretically analyzed. The validity of the analytical results is verified through extensive simulations, and they are in good agreement with each other. A useful guideline for step size selection to achieve a given EMSE is given, and major equations for convergence analysis and performance measures of the various S-LMS/S-LMM algorithms are also summarized in the form of a table in [36]. Computer simulations on system identification and joint adaptive noise cancellation (ANC) and AEC in automobiles with double-talk are conducted to verify the theoretical results and the effectiveness of the proposed algorithms. The rest of this paper is organized as follows. In Section II, the S-LMS family of algorithms are reviewed, and their robust counterparts are proposed. Their convergence performance analyses are given in Section III. The computer simulations are conducted in Section IV. Finally, the conclusions are drawn in Section V.

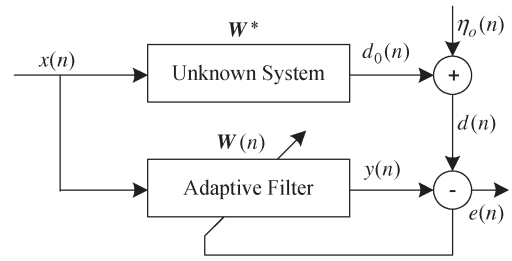


Fig. 1. Adaptive system identification.

II. S/SB/SPU-LMM/NLMM Algorithms

A. S-LMS Family of Algorithms

Consider the adaptive system identification problem in Fig. 1 where an input signal $x(n)$ is applied simultaneously to an L-order adaptive transversal filter with weight vector $\boldsymbol{W}(n) = [w_1(n), w_2(n), \ldots, w_L(n)]^T$ and an unknown system to be identified with an impulse response $\boldsymbol{W}^* = [w_1, w_2, \ldots, w_L]^T$. $\boldsymbol{X}(n) = [x(n), x(n-1), \ldots, x(n-L+1)]^T$ is the input vector. $(\cdot)^T$ denotes the transpose of a vector or a matrix. $e(n)$ is the estimation error.

The desired signal $d(n)$ of the adaptive filter is assumed to be corrupted by an additive noise $\eta_o(n)$

$$d(n) = \boldsymbol{X}^T(n)\boldsymbol{W}^* + \eta_o(n). \tag{1}$$

The update equations for the S-LMS family of algorithms can be written as

$$e(n) = d(n) - \boldsymbol{X}^T(n)\boldsymbol{W}(n) \tag{2}$$

$$\boldsymbol{W}(n+1) = \boldsymbol{W}(n) + \mu \boldsymbol{S}_{\boldsymbol{X}}(n)\boldsymbol{X}(n)e(n) \tag{3}$$

for the S/SB/SPU-LMS algorithms and

$$\boldsymbol{W}(n+1) = \boldsymbol{W}(n) + \mu \frac{\boldsymbol{S}_{\boldsymbol{X}}(n)\boldsymbol{X}(n)e(n)}{\varepsilon + \alpha \boldsymbol{X}^T(n)\boldsymbol{X}(n)} \tag{4}$$

TABLE I
LIST OF $\boldsymbol{S_X}(n)$ FOR THE S-LMS/LMM FAMILY OF ALGORITHMS

Algorithms	$\boldsymbol{S}_X(n)$
S-LMS, S-NLMS, S-LMM, S-NLMM	$s_i(n) = \begin{cases} 1, & (n+i) \bmod C = 0 \\ 0. & \text{otherwise} \end{cases}$
SB-LMS, SB-LMM, SB-NLMS, SB-NLMM	$\boldsymbol{S}_X(n) = \mathrm{diag}([\boldsymbol{s}_1^T(n), \boldsymbol{s}_2^T(n), \ldots, \boldsymbol{s}_C^T(n)]^T)$, $\boldsymbol{s}_j(n) = [s_{(j-1)\frac{L}{C}+1}, s_{(j-1)\frac{L}{C}+2}, \ldots, s_{j\frac{L}{C}}]^T,\ j = 1,2,\ldots,C$, $\boldsymbol{s}_j(n) = \begin{cases} [1,1,\ldots,1]^T, & n \bmod C + 1 = j \\ [0,0,\ldots,0]^T. & \text{otherwise} \end{cases}$
SPU-LMS, SPU-LMM, SPU-NLMS, SPU-NLMM	$r(n)$ is randomly selected from $[1,2,\ldots,C]$, $s_i(n) = \begin{cases} 1, & (n + r(n) + i) \bmod C = 0 \\ 0. & \text{otherwise} \end{cases}$

TABLE II
LIST OF COMPUTATIONAL COMPLEXITY OF THE S-LMS/LMM FAMILY OF ALGORITHMS (ATS)

Algorithms	Computational complexity (per. iteration)			
	Mult.	Add.	Div.	ATS
S/SB/SPU-LMS	$L+(L/C)$	$L+(L/C)$	0	0
S/SB/SPU-LMM	$L+(L/C)$	$L+(L/C)$	0	$O(N_w \log_2 N_w)$
S/SB/SPU-NLMS	$L+(L/C)+1$	$L+(L/C)+2$	1	0
S/SB/SPU-NLMM	$L+(L/C)+1$	$L+(L/C)+2$	1	$O(N_w \log_2 N_w)$

for the S/SB/SPU-NLMS algorithms, where μ is a constant step size parameter controlling the convergence rate and steady-state error of the algorithm. ε is a small positive value used to avoid division by zero, or it can be derived from some prior knowledge of the signal power. In the latter, α can serve as a weighting factor between the prior and instantaneous input signal power estimates. Otherwise, it is usually chosen to be one. $\boldsymbol{S_X}(n) = \mathrm{diag}(s_1(n), \ldots, s_L(n))$; $s_i(n) \in \{0, 1\}$, $i = 1, 2, \ldots, L$, is a diagonal selection matrix. At time instant n, when $s_i(n)$ is equal to one, the corresponding element $w_i(n)$ in $\boldsymbol{W}(n)$ will be updated. When $\boldsymbol{S_X}(n) = \boldsymbol{I}$, which is the identity matrix, (3) and (4) will reduce to the conventional LMS and NLMS algorithms, respectively. In the S-LMS algorithms, $\boldsymbol{W}(n)$ is divided into C nonoverlapping groups which are updated sequentially. The elements of $\boldsymbol{S_X}(n)$ are thus $P = L/C$ equally spaced or consecutive ones (and zeros elsewhere), and they are shifted cyclically as time propagates. Consequently, only P coefficients are updated per iteration. $\boldsymbol{S_X}(n)$ for the S-LMS family of algorithms and their computational complexities are summarized in Tables I and II, respectively.

B. S-LMM Family of Algorithms

Many techniques have been proposed to combat the adverse effects of impulsive noise on adaptive filters. They include the median-filtering algorithms [24], [25], the nonlinear clipping approaches [26], [27], and the approaches in [20] and [28] which are based on robust statistics [19]. The LMM [20] and the recursive least M-estimate (RLM) [28] algorithms are two effective algorithms derived by using robust M-estimation, and their improved robustness in impulsive noise and performance comparison with other relevant algorithms were thoroughly discussed in [20] and [28]. In this paper, we propose and study the convergence performance of an extension of the S-LMS family of algorithms using M-estimation for robust filtering in an impulsive noise environment. Next, we shall derive these algorithms which are referred to as the S-LMM family of algorithms.

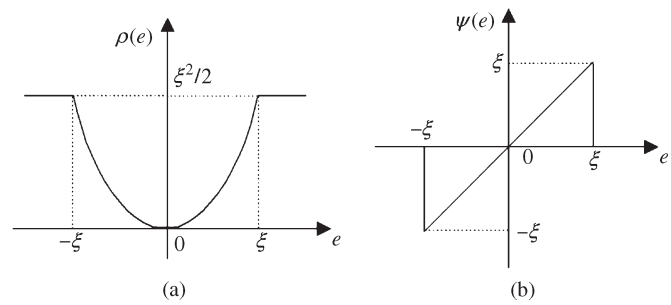


Fig. 2. (a) MH function $\rho(e)$. (b) $\psi(e)$, the score function of $\rho(e)$.

In the LMM algorithm, an M-estimate distortion measure $J_\rho = E[\rho(e(n))]$ is minimized, where $\rho(e)$, as shown in Fig. 2(a), is chosen as the MH function [19]

$$\rho(e) = \begin{cases} e^2/2, & 0 \le |e| < \xi \\ \xi^2/2, & \xi \le |e|. \end{cases} \tag{5}$$

ξ is a threshold parameter used to suppress the effect of outliers when the estimation error e is very large. Other M-estimate functions such as Hampel's three-part redescending function [29] can also be used. Notice that, when $\rho(e) = e^2/2$, it reduces to the conventional MSE criterion. Like the LMS algorithm, J_ρ is minimized by updating $\boldsymbol{W}(n)$ in the negative direction of the instantaneous gradient vector $\hat{\nabla}_{\boldsymbol{W}\rho}$. Therefore, the gradient vector $\nabla_{\boldsymbol{W}}(J_\rho)$ is approximated by $\hat{\nabla}_{\boldsymbol{W}\rho} = \partial\rho(e(n))/\partial\boldsymbol{W} = -\psi(e(n))\boldsymbol{X}(n)$, and $\psi(e) = \partial\rho(e)/\partial e$ is the score function, which is shown in Fig. 2(b). The following LMM algorithm can be obtained:

$$\boldsymbol{W}(n+1) = \boldsymbol{W}(n) - \mu\hat{\nabla}_{\boldsymbol{W}\rho} = \boldsymbol{W}(n) + \mu\psi(e(n))\,\boldsymbol{X}(n). \tag{6}$$

It can be seen that, when $|e(n)|$ is smaller than ξ, $\psi(e(n))$ is equal to $e(n)$, and (6) becomes identical to the LMS algorithm. When $|e(n)| > \xi$, $\psi(e(n))$ will become zero. Thus, the LMM algorithm can effectively reduce the adverse effect of the large estimation error on updating the filter coefficients. In the ATS method in [20] and [28], $e(n)$ is assumed to be Gaussian distributed except when being corrupted occasionally by additive impulsive noise, and the following robust variance estimate is proposed:

$$\hat{\sigma}_e^2(n) = \lambda_\sigma\hat{\sigma}_e^2(n-1) + c_1(1-\lambda_\sigma)\mathrm{med}\,(A_e(n)) \tag{7}$$

where λ_σ is a forgetting factor that is close to but smaller than one, $c_1 = 2.13$ is the finite sample correction factor, med$(\cdot)$ is the median operator, $A_e(n) = [e^2(n), \ldots, e^2(n - N_w + 1)]$, and N_w is the length of the data set. Using (7), the following adaptive threshold ξ can be obtained:

$$\xi = k_\xi\hat{\sigma}_e(n) \tag{8}$$

where k_ξ is a constant used to control the suppression of impulsive interference and a reasonable value of k_ξ is 2.576, and the window length N_w is usually chosen between five and nine [18]. In the case of a long sequence of impulses such as the double-talk encountered in AEC, the length of the median filter can be increased. If necessary, k_ξ can be increased slightly so that the AEC can track slow-varying channels while suppressing the double-talk. Following the same argument in deriving the LMM algorithm, we can extend the S-LMS family to the following:

$$\boldsymbol{W}(n+1) = \boldsymbol{W}(n) + \mu \boldsymbol{S}_{\boldsymbol{X}}(n)\psi\left(e(n)\right)\boldsymbol{X}(n) \tag{9}$$

for the S/SB/SPU-LMM algorithms and

$$\boldsymbol{W}(n+1) = \boldsymbol{W}(n) + \mu \frac{\boldsymbol{S}_{\boldsymbol{X}}(n)\psi\left(e(n)\right)\boldsymbol{X}(n)}{\varepsilon + \alpha \boldsymbol{X}^T(n)\boldsymbol{X}(n)} \tag{10}$$

for the S/SB/SPU-NLMM algorithms. The S-LMM family is rather similar to its S-LMS counterpart, except that $e(n)$ is now passed through a nonlinearity $\psi(e(n))$. The convergence performance of the LMS algorithm with error function nonlinearity was studied in [30]. Another related algorithm is the dual-sign LMS [31] algorithm. This was later studied in [26] and [32] by Koike and in [20], [28], and [33] using robust statistics. In [20] and [28], the threshold parameter ξ in the MH function is continuously updated as in (8), which greatly improves the convergence speed and steady-state error. Moreover, it has been found recently that the use of ATS and a class of nonlinearity called the M-nonlinearity can be conveniently analyzed [34]. The score function of the M-nonlinearity can be written as $\rho'_\xi(e) = \psi_\xi(e) = eq(e/\xi)$, where $q(e/\xi)$ is the score function and ξ is the threshold parameter. It covers most M-estimate functions and several commonly used error nonlinearities [34].

III. Mean and Mean Square Convergence Analyses

In this section, the convergence performance analysis of the S-LMS and S-LMM families of algorithms will be studied. In Section III-A1 hereinafter, the mean convergence analysis of the algorithm in Gaussian input and additive noise will be presented. We want to study under which condition will the mean weight vector converge to the desired Wiener solution. A difference equation in the mean weight-error vector is obtained from which we can determine the maximum step size for the mean weight vector to converge to the desired solution. Different special cases, such as the M-estimation- and LS-based algorithms, will be separately treated in the remarks. If the mean weight-error vector converges, the recursive algorithm is asymptotically unbiased with respect to the desired Wiener solution. To characterize the variations of the weight vector about its mean value, the mean square convergence analysis of the algorithm will be studied in Section III-A2 hereinafter. A difference equation involving the covariance matrices of the weight-error vector at successive time instants is derived from which we can study the condition on the step size for its convergence and the steady-state EMSE over the Wiener solution as a result of recursive adaptation. Two special cases, namely, the small step size and uncorrelated cases, will be studied in detail because of their importance and mathematical simplicity. In Section III-B, the analysis in Section III-A will be further extended to the case of Gaussian input with CG noise to model additive impulsive noise. The improvement of the M-estimate algorithms over the LS-based adaptive filters will be quantified theoretically. A summary of the major results is listed in [36].

The main contributions of the analysis include the following: 1) the use of Price's theorem [21], [22] to handle the nonlinearity variates in the algorithms and its extension [34] for the CG noise case; 2) the introduction of new special functions to decouple the difference equations describing the mean and mean square behaviors, and 3) the exploitation of the structural property of the selection matrix in the S-LMS family to evaluate the various important quantities such as step size bounds and EMSE. To simplify the analysis, we make the following assumptions, which are commonly used in the literature.

Assumption 1: The input signal $x(n)$ is an ergodic process which is Gaussian distributed with a zero mean and an autocorrelation matrix $\boldsymbol{R}_{\boldsymbol{XX}} = E[\boldsymbol{X}(n)\boldsymbol{X}^T(n)]$.

Assumption 2: The additive noise $\eta_o(n)$ is assumed to be a Gaussian noise $(\eta_o(n) = \eta_g(n))$ for the analysis in Section III-A hereinafter. For the analysis in Section III-B hereinafter, $\eta_o(n)$ is modeled as a CG noise [35] which is a frequently used model for analyzing impulsive noise. More precisely, it is given by

$$\eta_o(n) = \eta_g(n) + \eta_{im}(n) = \eta_g(n) + b(n)\eta_w(n) \tag{11}$$

where $\eta_g(n)$ and $\eta_w(n)$ are both independent and identically distributed (i.i.d.) zero-mean Gaussian sequences with variances σ_g^2 and σ_w^2, respectively. $b(n)$ is an i.i.d. Bernoulli random sequence whose value at any time instant is either zero or one, with occurrence probabilities $P_r(b(n)=1) = p_r$ and $P_r(b(n)=0) = 1 - p_r$. The variances of the random processes $\eta_{im}(n)$ and $\eta_o(n)$ are then given by $\sigma_{im}^2 = p_r\sigma_w^2$ and $\sigma_{\eta_o}^2 = \sigma_g^2 + \sigma_{im}^2 = \sigma_g^2 + p_r\sigma_w^2$, respectively. Moreover, we can see from (11) that the CG noise is equal to $\eta_o(n) = \eta_g(n)$ when $b(n) = 0$, and $\eta_o(n) = \eta_\Sigma(n) = \eta_g(n) + \eta_w(n)$ when $b(n) = 1$, with their respective variances given by σ_g^2 and $\sigma_\Sigma^2 = \sigma_g^2 + \sigma_w^2$. Accordingly, the probability distribution function of this CG distribution is given by

$$f_{\eta_o}(\eta) = \frac{1-p_r}{\sqrt{2\pi\sigma_g^2}}\exp\left(-\frac{\eta^2}{2\sigma_g^2}\right) + \frac{p_r}{\sqrt{2\pi\sigma_\Sigma^2}}\exp\left(-\frac{\eta^2}{2\sigma_\Sigma^2}\right). \tag{12}$$

It should be noted that the ratio $r_{im} = \sigma_{im}^2/\sigma_g^2 = p_r\sigma_w^2/\sigma_g^2$ is a measure of the impulsive characteristic of the CG noise.

Assumption 3: $\boldsymbol{W}(n)$, $x(n)$, and $\eta_o(n)$ are statistically independent (the independence assumption [6]). It is a good approximation for a large value of L and is commonly used to simplify the convergence analysis of numerous adaptive filtering algorithms. Interested readers are referred to [13] for a comparison of the results obtained from the independence assumption and the general case of mixing signals for the SPU-LMS algorithm. The Gaussian input assumption in

Assumption 1 makes the problem mathematically tractable and allows accurate close-form expressions to be obtained in many important cases. Finally, we denote $\boldsymbol{W}^* = \boldsymbol{R}_{\boldsymbol{XX}}^{-1}\boldsymbol{P}_{dX}$ as the optimal Wiener solution, where $\boldsymbol{P}_{d\boldsymbol{X}} = E[d(n)\boldsymbol{X}(n)]$ is the ensemble-averaged cross-correlation vector between $\boldsymbol{X}(n)$ and $d(n)$.

A. Mean and Mean Square Behaviors in Gaussian Noise

1) Mean Behavior: From (10), the weight-error vector $\boldsymbol{v}(n) = \boldsymbol{W}^* - \boldsymbol{W}(n)$ for the S/SB/SPU-NLMM algorithms can be written as

$$\boldsymbol{v}(n+1) = \boldsymbol{v}(n) - \frac{\mu \boldsymbol{S}_{\boldsymbol{X}}(n)\psi\left(e(n)\right)\boldsymbol{X}(n)}{\varepsilon + \alpha \boldsymbol{X}^T(n)\boldsymbol{X}(n)} \tag{13}$$

where $\boldsymbol{W}^*$ is the optimal weight vector previously defined. $\psi(e(n))$ is a clipping nonlinearity, which is usually chosen from the M-nonlinearity family [34]. When it is equal to $e(n)$, (13) reduces to the conventional S/SB/SPU-NLMS algorithms. Taking the expectation over $\{\boldsymbol{v}, \boldsymbol{X}, \eta_g\}$ on both sides of (13) yields

$$E\left[\boldsymbol{v}(n+1)\right] = E\left[\boldsymbol{v}(n)\right] - \mu\widetilde{\boldsymbol{H}} \tag{14}$$

where $E[\cdot]$ denotes the expectation over $\{\boldsymbol{v}(n), \boldsymbol{X}(n), \eta_g(n)\}$ (also written as $E_{\{\boldsymbol{v},\boldsymbol{X},\eta_g\}}[\cdot]$ for clarity), and $\widetilde{\boldsymbol{H}} = E_{\{\boldsymbol{v},\boldsymbol{X},\eta_g\}}[\boldsymbol{S}_{\boldsymbol{X}}(n)\psi(e(n))\boldsymbol{X}(n)/(\varepsilon + \alpha\boldsymbol{X}^T(n)\boldsymbol{X}(n))]$. Dropping the time index of $\boldsymbol{S}_{\boldsymbol{X}}(n)$, $\boldsymbol{X}(n)$, $e(n)$, and $\eta_g(n)$, one gets

$$\widetilde{\boldsymbol{H}} = E_{\{\boldsymbol{v},\boldsymbol{X},\eta_g\}}\left[\boldsymbol{S}_{\boldsymbol{X}}\psi(e)\boldsymbol{X}/(\varepsilon + \alpha\boldsymbol{X}^T\boldsymbol{X})\right] = E_{\{\boldsymbol{v}\}}[\boldsymbol{H}_1] \tag{15}$$

where $\boldsymbol{H}_1 = E_{\{\boldsymbol{X},\eta_g\}}[\boldsymbol{S}_{\boldsymbol{X}}\psi(e)\boldsymbol{X}/(\varepsilon + \alpha\boldsymbol{X}^T\boldsymbol{X})|\boldsymbol{v}]$ and the second equation is obtained from the independence assumption of $\eta_g(n)$, $\boldsymbol{W}(n)$, and $x(n)$ in Assumption 3.

All the algorithms in the S-LMS/LMM family share a common structural property: There are C different and independent combinations for $\boldsymbol{S}_{\boldsymbol{X}}$, which are denoted by $\boldsymbol{S}_{\boldsymbol{X}}^{(i)}$, $i = 1, 2, \ldots, C$. Each of them possesses an equal occurrence probability of $p_i = 1/C$. It then follows that $\Sigma_{i=1}^{C}\boldsymbol{S}_{\boldsymbol{X}}^{(i)} = \boldsymbol{I}$. Since $\boldsymbol{X}$ is stationary and $\boldsymbol{S}_{\boldsymbol{X}}^{(i)}$ is independent, we get

$$\begin{aligned}\boldsymbol{H}_1 &= \Sigma_{i=1}^{C} p_i E_{\{\boldsymbol{X},\eta_g\}}\left[\boldsymbol{S}_{\boldsymbol{X}}^{(i)}\psi(e)\boldsymbol{X}/(\varepsilon + \alpha\boldsymbol{X}^T\boldsymbol{X})|\boldsymbol{v}\right] \\ &= \left(\Sigma_{i=1}^{C}\boldsymbol{S}_{\boldsymbol{X}}^{(i)}\right)\cdot p_i\boldsymbol{H} \\ &= \frac{1}{C}\boldsymbol{H}\end{aligned} \tag{16}$$

where $\boldsymbol{H} = E_{\{\boldsymbol{X},\eta_g\}}[\psi(e)\boldsymbol{X}/(\varepsilon + \alpha\boldsymbol{X}^T\boldsymbol{X})|\boldsymbol{v}]$ is evaluated in [36, App. A] to be

$$\boldsymbol{H} \approx \overline{\psi'}\left(\sigma_e^2(n)\right)\boldsymbol{U}\Lambda\boldsymbol{D}_\Lambda\boldsymbol{U}^T\boldsymbol{v}(n) \tag{17}$$

where $\overline{\psi'}\left(\sigma_e^2\right) = \int_{-\infty}^{\infty}\left(\psi'(e)/\sqrt{2\pi}\sigma_e\right)\exp(-(e^2/2\sigma_e^2))de$, $\sigma_e^2(n) = \boldsymbol{v}^T(n)\boldsymbol{R}_{\boldsymbol{XX}}\boldsymbol{v}(n) + \sigma_g^2$, $\boldsymbol{R}_{\boldsymbol{XX}} = \boldsymbol{U}\Lambda\boldsymbol{U}^T$ is the eigenvalue decomposition of $\boldsymbol{R}_{\boldsymbol{XX}}$, $\boldsymbol{U}$ is some orthogonal matrix, and $\Lambda = \mathrm{diag}(\lambda_1, \lambda_2, \ldots, \lambda_L)$ contains the corresponding eigenvalues. $\boldsymbol{D}_\Lambda$ is a diagonal matrix with its ith diagonal entry given by the following Abelian integral function [34], [36]:

$$\begin{aligned}[\boldsymbol{D}_\Lambda]_{i,i} &= I_i(\Lambda) \\ &= \int_0^\infty \exp(-\beta\varepsilon)\left[\prod_{k=1}^{L}(2\alpha\beta\lambda_k + !1)^{-1/2}\right](2\alpha\beta\lambda_i + 1)^{-1}\,d\beta.\end{aligned} \tag{18}$$

For a given $\psi(e)$, $\overline{\psi'}(\sigma_e^2)$ can be evaluated analytically or numerically. Substituting (15)–(17) into (14), the following difference equation in the mean weight-error vector is obtained:

$$E\left[\boldsymbol{v}(n+1)\right] = \left(\boldsymbol{I} - \frac{\mu}{C}A_\psi\left(\sigma_e^2(n)\right)\boldsymbol{U}\Lambda\boldsymbol{D}_\Lambda\boldsymbol{U}^T\right)E\left[\boldsymbol{v}(n)\right] \tag{19}$$

where $\overline{\psi'}(\sigma_e^2)$ is simply written as $A_\psi(\sigma_e^2)$. $\boldsymbol{D}_\Lambda$ and $A_\psi(\sigma_e^2)$ result from the use of normalization and nonlinearity, respectively. For the conventional LMS algorithm, $\boldsymbol{D}_\Lambda$ and $A_\psi(\sigma_e^2)$ are equal to the identity matrix and one, respectively. For notation convenience, we also use $\sigma_e^2(n)$ and σ_e^2 interchangeably and replace the approximate sign in (17) by the equality sign. Equation (19) can also be written in the natural coordinate $\boldsymbol{V}(n) = \boldsymbol{U}^T\boldsymbol{v}(n)$ as

$$E\left[\boldsymbol{V}(n+1)\right] = \left(\boldsymbol{I} - \frac{\mu}{C}A_\psi\left(\sigma_e^2(n)\right)\Lambda\boldsymbol{D}_\Lambda\right)E\left[\boldsymbol{V}(n)\right]. \tag{20}$$

This is equivalent to the following L scalar first-order finite difference equations:

$$E\left[\boldsymbol{V}(n+1)\right]_i = \left(1 - \frac{\mu}{C}A_\psi\left(\sigma_e^2(n)\right)\lambda_i I_i(\Lambda)\right)E\left[\boldsymbol{V}(n)\right]_i \tag{21}$$

where $E[\boldsymbol{V}(n)]_i$ is the ith element of the vector $E[\boldsymbol{V}(n)]$ for $i = 1, 2, \ldots, L$. We discuss the LMS-based algorithm and the LMM-based algorithms separately in the following remarks.

Remarks: (R-A1): **S/SB/SPU-LMS/NLMS algorithms**: For the conventional S/SB/SPU-LMS/NLMS algorithms, $\psi(e) = e$, and $\overline{\psi'}(\sigma_e^2) = A_\psi(\sigma_e^2) = 1$. In particular, when $\boldsymbol{D}_\Lambda = \boldsymbol{I}$, the analysis reduces to the S/SB/SPU-LMS algorithms. The mean weight vector of the S/SB/SPU-NLMS algorithms will converge if

$$\left|1 - \frac{\mu}{C}\lambda_i I_i(\Lambda)\right| < 1, \qquad \text{for all } i\text{'s} \tag{22}$$

and the corresponding maximum step size satisfies

$$\mu_{\max} < 2C/\left(\lambda_i I_i(\Lambda)\right), \qquad \text{for all } i. \tag{23}$$

It can be shown in [33] that the maximum value of $\lambda_i I_i(\Lambda)$ occurs also at the largest eigenvalue $\lambda_{\max}$. Denoting the corresponding value of $I_i(\Lambda)$ by $I_{i_\lambda_{\max}}(\Lambda)$, we get the following bound for the maximum possible step size for the convergence of the mean weight-error vector:

$$\mu_{\max} < 2C/\left(\lambda_{\max} I_{i_\lambda_{\max}}(\Lambda)\right). \tag{24}$$

Compared with the LMS-based algorithms, the step size of the normalized algorithms is changed by a factor $1/I_{i_\lambda_{\max}}(\Lambda)$.

TABLE III
LIST OF $A_\psi(\sigma_e^2)$, $B_\psi(\sigma_e^2)$, AND $C_\psi(\sigma_e^2)$ FOR RELATED ALGORITHMS

Non-linearity	$\psi(e)$	$A_\psi(\sigma_e^2)=\overline{\psi'}(\sigma_e^2)$	$B_\psi(\sigma_e^2)=E[\psi^2(e)]$	$C_\psi(\sigma_e^2)=\frac{d}{d\sigma_e^2}E[\psi^2(e)]$
Modified Huber	$\psi_{\mathrm{MH}}(\varepsilon)=\begin{cases}\varepsilon, & \lvert\varepsilon\rvert\le\xi\\ 0, & \text{otherwise}\end{cases}$	$\frac{2}{\sqrt{2\pi}\sigma_e}[\int_0^\xi \exp(-\frac{e^2}{2\sigma_e^2})de-\xi\exp(-\frac{\xi^2}{2\sigma_e^2})]$	$\frac{2\sigma_e}{\sqrt{2\pi}}[\int_0^\xi \exp(-\frac{e^2}{2\sigma_e^2})de-\xi\exp(-\frac{\xi^2}{2\sigma_e^2})]$	$\frac{2}{\sqrt{2\pi}\sigma_e}[\int_0^\xi \exp(-\frac{e^2}{2\sigma_e^2})de-\xi\exp(-\frac{\xi^2}{2\sigma_e^2})]$ $-(\frac{\xi^3}{\sqrt{2\pi}\sigma_e^3})\exp(-\frac{\xi^2}{2\sigma_e^2})$
Error function [30]	$\psi_{\mathrm{EF}}(e)=\int_0^e \exp(-u^2/2\sigma_y^2)du$	$\sigma_y/(\sigma_y^2+\sigma_e^2)^{1/2}$	$\sigma_y^2\sin^{-1}(\frac{1}{1+(\sigma_y^2/\sigma_e^2)})$	$[(1+2\sigma_e^2/\sigma_y^2)(1+\sigma_e^2/\sigma_y^2)]^{-1}$
Dual sign [31]	$\psi_{\mathrm{DS}}(e)=\begin{cases}\mathrm{sgn}(e), & \lvert e\rvert\le\tau\\ L\,\mathrm{sgn}(e), & \lvert e\rvert>\tau\end{cases}$	$\sqrt{\frac{2}{\pi}}\frac{1}{\sigma_e}[1+(L-1)\exp(-\tau^2/(2\sigma_e^2))]$	$L^2-(L^2-1)\sqrt{\frac{2}{\pi}}\int_0^{\tau/\sigma_e^2}\exp(-u^2/2)du$	$\frac{1}{2}\exp(-(\tau^2/2\sigma_e^3))(L^2-1)(\tau/\sigma_e^3)$
Quantizer [38]	$\psi_{\mathrm{Q}}(\mu e)=\begin{cases}0 & \lvert\mu e\rvert<\Delta/2\\ \Delta\,\mathrm{sgn}(\mu e) & \Delta/2\le\lvert\mu e\rvert<\Delta\\ \mu e & \text{otherwise}\end{cases}$ (Δ is the size of the dead zone)	$\mu[1+\sqrt{\frac{2}{\pi}}\frac{\Delta}{\mu\sigma_e}\exp(\frac{-\Delta^2}{8\mu^2\sigma_e^2})$ $-\mathrm{erf}(\frac{\Delta}{\sqrt{2}\mu\sigma_e})]$	$\mu^2\sigma_e^2\{(\frac{\Delta}{\mu\sigma_e})^2[\mathrm{erf}(\frac{\Delta}{\sqrt{2}\mu\sigma_e})-\mathrm{erf}(\frac{\Delta}{2\sqrt{2}\mu\sigma_e})]$ $+1+\sqrt{\frac{2}{\pi}}(\frac{\Delta}{\mu\sigma_e})\exp(\frac{-\Delta^2}{2\mu^2\sigma_e^2})-\mathrm{erf}(\frac{\Delta}{\sqrt{2}\mu\sigma_e})\}$	---

When $\mu = C/(\lambda_{\max} I_{i_\lambda_{\max}}(\Lambda))$, the fastest convergence rate is

$$1-\frac{2\lambda_{\min} I_{i_\lambda_{\min}}(\Lambda)}{\lambda_{\max} I_{i_\lambda_{\max}}(\Lambda)}. \tag{25}$$

It is limited by the mode corresponding to the smallest eigenvalue $\lambda_{\min}$ with the corresponding value of $I_i(\Lambda)$ given by $I_{i_\lambda_{\min}}(\Lambda)$. The smaller the value in (25), the faster the convergence rate will be. From the definition of $I_i(\Lambda)$, it can be shown that $I_{i_\lambda_{\min}}(\Lambda)/I_{i_\lambda_{\max}}(\Lambda)\ge 1$. In other words, the eigenvalue spread $\lambda_{\max}/\lambda_{\min}$ is reduced by a factor $I_{i_\lambda_{\max}}(\Lambda)/I_{i_\lambda_{\min}}(\Lambda)$ after the normalization. Therefore, under the stated assumptions, the convergence rate of the normalized algorithms will always be faster than their LMS counterparts if the eigenvalues are unequal. When $C=1$, (23) reduces to the conventional LMS and NLMS algorithms [34], [37]. Compared with them, the convergence rate of the sequential PU algorithms is decreased by a factor of C for the same step size μ, which also agrees with the result for the SPU-LMS algorithm in [13, eq. (11)]. As conclude in [13], the SPU-LMS algorithm is more attractive than the S/SB-LMS algorithms, which may diverge for certain signals due to the predetermined updating strategy. A similar argument applies to the SPU-NLMS algorithm.

(R-A2): **M-nonlinearity and the LMM/NLMM-based algorithms**: For general M-nonlinearity other than $\psi(e)=e$, (19) or (20) becomes a set of nonlinear difference equations. A general solution is rather difficult to obtain because the term $A_\psi(\sigma_e^2)$ is dependent on the MSE. For $\boldsymbol{D}_\Lambda=\boldsymbol{I}$, $C=1$, (i.e., the LMS-based algorithms) and the dual-sign nonlinearity, (21) agrees with the result in [31]. When the error function nonlinearity [30] is considered, (19) also agrees with the result in [30]. The case for the NLMS and NLMM algorithms with general M-nonlinearity was studied in [34]. It was found that (21) provides a good approximation at the steady state of the normalized algorithms and for the S-NLMM algorithm with ATS. It should be noted that no such approximation is used in the variants of the LMS algorithms. Overall, we can see that the sequential updating will reduce the effective step size by a factor of C since the weight vector is only partially updated in a sequential manner with a period of C. Similar to the conclusion obtained in [34], if $A_{\mathrm{MH}}(\sigma_e^2)$ is not made adaptive, an inappropriately chosen ξ may lead to a very small value of $A_{\mathrm{MH}}(\sigma_e^2)$ and, hence, of the step size, leading to slow adaptation. For the ATS in (8) and the M-nonlinearity, $A_{\mathrm{MH}}(\sigma_e^2)$ is maintained approximately at $A_{\mathrm{MH}}(\sigma_e^2)\approx \mathrm{erf}(k_\xi/\sqrt{2})-(2k_\xi/\sqrt{2\pi})\exp(-(k_\xi^2/2))$ (if $\hat{\sigma}_e^2\approx\sigma_e^2$), which is a constant and is slightly less than one. The degradation in the convergence rate over their LMS/NLMS-based counterparts is therefore minimal. Interested readers are referred to [34] for more details.

Although the maximum possible step size is, in general, difficult to obtain for arbitrary nonlinearity, a sufficient condition for the mean weight vector of the algorithm to converge is $|1-(\mu/C)A_\psi(\sigma_e^2)\lambda_i I_i(\Lambda)|<1$, for all i's. If $\overline{\psi'}(\sigma_e^2)$ is bounded above by a constant $A_{\psi_\max}$, then a conservative maximum step size is

$$\mu_{\max} < 2C/\left(A_{\psi_\max}\lambda_{\max} I_{i_\lambda_{\max}}(\Lambda)\right) \tag{26}$$

which yields good estimates in practical algorithms.

If $\mu\psi(e(n))$ in (13) is replaced by $\psi_Q(\mu e(n))$, where $\psi_Q(\cdot)$ is a quantizer function, then the aforementioned analysis can be used to model finite word length effects of the algorithms as had been done for the LMS algorithm in [38]. The corresponding values of $A_\psi(\sigma_e^2)$ and $B_\psi(\sigma_e^2)$ for $\psi_Q(\cdot)$ are summarized in Table III. Due to page limitation, the usefulness of this model in quantifying the finite word length effect of the S-LMS family is reported in [36].

2) Mean Square Behavior: By postmultiplying (13) by its transpose and taking the expectation over $\{\boldsymbol{v},\boldsymbol{X},\eta_g\}$, we obtain a difference equation involving the covariance matrices of the weight-error vector at successive time instants

$$\boldsymbol{\Xi}(n+1)=\boldsymbol{\Xi}(n)-\boldsymbol{M}_1-\boldsymbol{M}_2+\boldsymbol{M}_3 \tag{27}$$

where $\boldsymbol{\Xi}(n)=E[\boldsymbol{v}(n)\boldsymbol{v}^T(n)]$

$$\begin{aligned}\boldsymbol{M}_1 &= \mu E_{\{\boldsymbol{v}\}}\left[E_{\{\boldsymbol{X},\eta_g\}}\left[\frac{\boldsymbol{S}_{\boldsymbol{X}}\psi(e)\boldsymbol{X}}{\varepsilon+\alpha\boldsymbol{X}^T\boldsymbol{X}}|\boldsymbol{v}\right]\boldsymbol{v}^T\right]\\ &= \mu E_{\{\boldsymbol{v}\}}[\boldsymbol{H}_1\boldsymbol{v}^T]\\ &\approx \frac{\mu}{C}A_\psi\left(\sigma_e^2\right)\boldsymbol{U}\Lambda\boldsymbol{D}_\Lambda\boldsymbol{U}^T\boldsymbol{\Xi}(n)\end{aligned} \tag{28}$$

$$\begin{aligned}\boldsymbol{M}_2 &= \boldsymbol{M}_1^T \\ &= \mu E_{\{\boldsymbol{v}\}}\left[\boldsymbol{v}\boldsymbol{H}_1^T\right] \\ &\approx \frac{\mu}{C} A_\psi\left(\sigma_e^2\right)\boldsymbol{\Xi}(n)\boldsymbol{U}\boldsymbol{D}_\Lambda\Lambda\boldsymbol{U}^T \end{aligned} \tag{29}$$

$$\begin{aligned}\boldsymbol{M}_3 &= E_{\{\boldsymbol{v},\boldsymbol{X},\eta_g\}}\left[\frac{\mu^2\psi^2(e)}{\left(\varepsilon+\alpha\boldsymbol{X}^T\boldsymbol{X}\right)^2}\boldsymbol{S}_{\boldsymbol{X}}\boldsymbol{X}\boldsymbol{X}^T\boldsymbol{S}_{\boldsymbol{X}}\right] \\ &= \mu^2 E_{\{\boldsymbol{v}\}}\left[E_{\{\boldsymbol{X},\eta_g\}}\left[\frac{\psi^2(e)\boldsymbol{S}_{\boldsymbol{X}}\boldsymbol{X}\boldsymbol{X}^T\boldsymbol{S}_{\boldsymbol{X}}}{\left(\varepsilon+\alpha\boldsymbol{X}^T\boldsymbol{X}\right)^2}|\boldsymbol{v}\right]\right] \\ &= \mu^2 E_{\{\boldsymbol{v}\}}[\widetilde{\boldsymbol{s}}_3]. \end{aligned} \tag{30}$$

Note that the final expressions in (28) and (29) are obtained by using our previous results in (15)–(17). $\widetilde{\boldsymbol{s}}_3$ is the expectation of $\psi^2(e)\boldsymbol{S}_{\boldsymbol{X}}\boldsymbol{X}\boldsymbol{X}^T\boldsymbol{S}_{\boldsymbol{X}}/(\varepsilon+\alpha\boldsymbol{X}^T\boldsymbol{X})^2$ taken over $\{\boldsymbol{X},\eta_g\}$ conditioned on $\boldsymbol{v}$. Using a similar approach as that previously mentioned yields

$$\begin{aligned}\widetilde{\boldsymbol{s}}_3 &= \sum_{i=1}^{C} p_i E_{\{\boldsymbol{X},\eta_g\}}\left[\frac{\psi^2(e)\boldsymbol{S}_{\boldsymbol{X}}^{(i)}\boldsymbol{X}\boldsymbol{X}^T\boldsymbol{S}_{\boldsymbol{X}}^{(i)}}{\left(\varepsilon+\alpha\boldsymbol{X}^T\boldsymbol{X}\right)^2}|\boldsymbol{v}\right] \\ &= \sum_{i=1}^{C} p_i \boldsymbol{S}_{\boldsymbol{X}}^{(i)}\boldsymbol{s}_3\boldsymbol{S}_{\boldsymbol{X}}^{(i)} \\ &= \frac{1}{C}(\boldsymbol{s}_3\circ\Omega) \end{aligned} \tag{31}$$

where $\boldsymbol{s}_3 = E_{\{\boldsymbol{X},\eta_g\}}[\psi^2(e)\boldsymbol{X}\boldsymbol{X}^T/(\varepsilon+\alpha\boldsymbol{X}^T\boldsymbol{X})^2|\boldsymbol{v}]$ and the last equation is obtained by using the identity $\sum_{i=1}^{C} p_i\boldsymbol{S}_{\boldsymbol{X}}^{(i)}(\boldsymbol{B})\boldsymbol{S}_{\boldsymbol{X}}^{(i)} = (1/C)\boldsymbol{B}\circ\Omega$, where $\boldsymbol{B}$ is any matrix with an appropriate dimension and $\circ$ is the elementwise (or Hadamard) product of matrices. For the S/SPU-LMS/LMM/NLMS/NLMM algorithms, one can verify from $\boldsymbol{S}_{\boldsymbol{X}}(n)$ in Table I that $\Omega = (\boldsymbol{1}_{L/C}\otimes\boldsymbol{I})$, and for the SB-LMS/LMM/NLMS/NLMM algorithms, $\Omega = (\boldsymbol{I}\otimes\boldsymbol{1}_{L/C})$, where $\boldsymbol{1}_{L/C}$ is an $(L/C)\times(L/C)$ matrix with all entries equal to one, $\otimes$ denotes the Kronecker product, and $\boldsymbol{I}$ is the $C\times C$ identity matrix. It is shown in [36, App. B] that

$$\boldsymbol{s}_3 \approx 2C_\psi\left(\sigma_e^2\right)\boldsymbol{U}\left[\Lambda\left(\boldsymbol{V}\boldsymbol{V}^T\circ\boldsymbol{I}(\Lambda)\right)\Lambda\right]\boldsymbol{U}^T + B_\psi\left(\sigma_e^2\right)\boldsymbol{U}\Lambda\boldsymbol{I}'(\Lambda)\boldsymbol{U}^T \tag{32}$$

where $B_\psi(\sigma_e^2) = E[\psi^2(e)] = (1/\sqrt{2\pi}\sigma_e)\int_{-\infty}^{\infty}\psi^2(e)\exp(-(e^2/2\sigma_e^2))de$, $C_\psi(\sigma_e^2) = (d/d\sigma_e^2)E[\psi^2(e)]$, and $\boldsymbol{V} = \boldsymbol{U}^T\boldsymbol{v}$. $\boldsymbol{I}(\Lambda)$ is an $L\times L$ matrix, and its (i,j)th element $[\boldsymbol{I}(\Lambda)]_{ij}$ is

$$\begin{aligned}I_{ij}(\Lambda) = \int_0^\infty \beta\exp(-\beta\varepsilon)\left[\prod_{k=1}^{L}(2\alpha\beta\lambda_k+1)^{-1/2}\right] \\ \cdot(2\alpha\beta\lambda_i+1)^{-1}(2\alpha\beta\lambda_j+1)^{-1}d\beta \end{aligned} \tag{33}$$

and $\boldsymbol{I}'(\Lambda)$ is a diagonal matrix whose ith diagonal element is

$$I_i'(\Lambda) = \int_0^\infty \beta\exp(-\beta\varepsilon)\left[\prod_{k=1}^{L}(2\alpha\beta\lambda_k+1)^{-\frac{1}{2}}\right](2\alpha\beta\lambda_i+1)^{-1}d\beta. \tag{34}$$

$I_{ij}(\Lambda)$ and $I_i'(\Lambda)$ are another two Abelian integral functions, and they, together with $\boldsymbol{D}_\Lambda$, account for the effect of normalization. For more details, see [34] and [36].

For a given nonlinearity $\psi(e)$, $B_\psi(\sigma_e^2)$ and $C_\psi(\sigma_e^2)$ can be computed analytically or numerically. Substituting (28)–(32) into (27), we have

$$\begin{aligned}\boldsymbol{\Xi}(n+1) \approx \boldsymbol{\Xi}(n) &- \frac{\mu}{C}A_\psi\left(\sigma_e^2\right)\boldsymbol{U}\Lambda\boldsymbol{D}_\Lambda\boldsymbol{U}^T\boldsymbol{\Xi}(n) \\ &- \frac{\mu}{C}A_\psi\left(\sigma_e^2\right)\boldsymbol{\Xi}(n)\boldsymbol{U}\boldsymbol{D}_\Lambda\Lambda\boldsymbol{U}^T + \frac{2\mu^2}{C}C_\psi\left(\sigma_e^2\right) \\ &\times\left\{\boldsymbol{U}\Lambda\left[\left(\boldsymbol{U}^T\boldsymbol{\Xi}(n)\boldsymbol{U}\right)\circ\boldsymbol{I}(\Lambda)\right]\Lambda\boldsymbol{U}^T\right\}\circ\Omega \\ &+ \frac{\mu^2}{C}B_\psi\left(\sigma_e^2\right)\left(\boldsymbol{U}\Lambda\boldsymbol{I}'(\Lambda)\boldsymbol{U}^T\right)\circ\Omega. \end{aligned} \tag{35}$$

For the S/SB/SPU-LMS/LMM algorithms, $\boldsymbol{I}'(\Lambda) = \boldsymbol{D}_\Lambda = \boldsymbol{I}$, and $[\boldsymbol{I}(\Lambda)]_{ij} = I_{ij}(\Lambda) = 1$ for all i's and j's (i.e., $\boldsymbol{I}(\Lambda) = \boldsymbol{1}_L$). Equation (35) reduces to

$$\begin{aligned}\boldsymbol{\Xi}(n+1) \approx \boldsymbol{\Xi}(n) &- \frac{\mu}{C}A_\psi\left(\sigma_e^2\right)\boldsymbol{R}_{\boldsymbol{X}\boldsymbol{X}}\boldsymbol{\Xi}(n) \\ &- \frac{\mu}{C}A_\psi\left(\sigma_e^2\right)\boldsymbol{\Xi}(n)\boldsymbol{R}_{\boldsymbol{X}\boldsymbol{X}} \\ &+ \frac{2\mu^2}{C}C_\psi\left(\sigma_e^2\right)\left(\boldsymbol{R}_{\boldsymbol{X}\boldsymbol{X}}\boldsymbol{\Xi}(n)\boldsymbol{R}_{\boldsymbol{X}\boldsymbol{X}}\right)\circ\Omega \\ &+ \frac{\mu^2}{C}B_\psi\left(\sigma_e^2\right)\boldsymbol{R}_{\boldsymbol{X}\boldsymbol{X}}\circ\Omega. \end{aligned} \tag{36}$$

Equation (35) can be simplified further by using the natural coordinate $\boldsymbol{\Phi}(n) = \boldsymbol{U}^T\boldsymbol{\Xi}(n)\boldsymbol{U}$ to

$$\begin{aligned}\boldsymbol{\Phi}(n+1) \approx \boldsymbol{\Phi}(n) &- \frac{\mu}{C}A_\psi\left(\sigma_e^2\right)\Lambda\boldsymbol{D}_\Lambda\boldsymbol{\Phi}(n) \\ &- \frac{\mu}{C}A_\psi\left(\sigma_e^2\right)\boldsymbol{\Phi}(n)\boldsymbol{D}_\Lambda\Lambda + \frac{2\mu^2}{C}C_\psi\left(\sigma_e^2\right)\boldsymbol{U}^T \\ &\times\left\{\left[\boldsymbol{U}\Lambda\left(\boldsymbol{\Phi}(n)\circ\boldsymbol{I}(\Lambda)\right)\Lambda\boldsymbol{U}^T\right]\circ\Omega\right\}\boldsymbol{U} \\ &+ \frac{\mu^2}{C}B_\psi\left(\sigma_e^2\right)\boldsymbol{U}^T\left[\left(\boldsymbol{U}\Lambda\boldsymbol{I}'(\Lambda)\boldsymbol{U}^T\right)\circ\Omega\right]\boldsymbol{U}. \end{aligned} \tag{37}$$

Due to the matrix Ω, the difference equation is considerably more complicated than the conventional LMS/NLMS algorithm. If $\boldsymbol{R}_{\boldsymbol{X}\boldsymbol{X}}$ is block diagonal with the same structure as Ω for the LMS algorithm, $A_\psi(\sigma_e^2)$, $S_\psi(\sigma_e^2) = B_\psi(\sigma_e^2)/\sigma_e^2$, and $C_\psi(\sigma_e^2)$ are all equal to one, $\boldsymbol{I}'(\Lambda) = \boldsymbol{D}_\Lambda = \boldsymbol{I}$, and $\boldsymbol{I}(\Lambda) = \boldsymbol{1}_L$, then (37) will reduce to [13, eq. (12)] after taking into account the average behavior of $\boldsymbol{S}_{\boldsymbol{X}}(n)$. Since Ω in this case will not change the other terms in (37), it can be treated as the identity matrix. For the NLMS algorithm, (37) will then reduce to the difference equation similar to that of the NLMS case with M-nonlinearity [34], [37] except for the decimation factor C. Similarly, the EMSE can be derived as $\text{EMSE}(n) = \text{Tr}(\boldsymbol{R}_{\boldsymbol{X}\boldsymbol{X}}\boldsymbol{\Xi}(n)) = \text{Tr}(\Lambda\boldsymbol{\Phi}(n))$[6]. In general, to analyze the mean square convergence condition, we can write $\boldsymbol{\Phi}(n)$ using the vec$(\cdot)$ notation and obtain a difference equation in $\Theta(n) = \text{vec}(\boldsymbol{\Phi}(n))$. A step size bound for convergence can be estimated. The technical details are elaborated in [36, App. C]. We now analyze a few interesting cases hereinafter.

Small Step Sizes: From the definition of the EMSE mentioned earlier, the steady-state EMSE can be computed as $\text{EMSE}(\infty) = \text{Tr}(\Lambda\boldsymbol{\Phi}(\infty)) = \text{Tr}(\Lambda\boldsymbol{\Phi}_D(\infty))$, where $\boldsymbol{\Phi}_D(\infty)$

is a diagonal matrix whose diagonal elements are those of $\boldsymbol{\Phi}(\infty)$. If a small EMSE is required, μ is usually very small, and therefore, the fourth term on the right-hand side of (37) is negligible. At the steady state, $n \to \infty$, and after some manipulations to extract the diagonal elements of $\boldsymbol{\Phi}(\infty)$, we obtain

$$\boldsymbol{\Phi}_D(\infty) \approx \mu \frac{B_\psi\left(\sigma_e^2(\infty)\right)}{2A_\psi\left(\sigma_e^2(\infty)\right)} \Lambda^{-1} \boldsymbol{D}_\Lambda^{-1} \times \left\{\left[\boldsymbol{U}^T\left(\left(\boldsymbol{U}\Lambda \boldsymbol{I}'(\Lambda)\boldsymbol{U}^T\right)\circ\Omega\right)\boldsymbol{U}\right]\circ \boldsymbol{I}\right\}. \quad (38)$$

Hence, the steady-state EMSE is given by

$$\mathrm{EMSE}(\infty) \approx \mu \frac{B_\psi\left(\sigma_e^2(\infty)\right)}{2A_\psi\left(\sigma_e^2(\infty)\right)} \times \mathrm{Tr}\left(\boldsymbol{D}_\Lambda^{-1}\boldsymbol{U}^T\left[\left(\boldsymbol{U}\Lambda\boldsymbol{I}'(\Lambda)\boldsymbol{U}^T\right)\circ\Omega\right]\boldsymbol{U}\right). \quad (39)$$

For clarity, appropriate subscripts will be appended to the symbol $\mathrm{EMSE}(\infty)$ to differentiate the steady-state EMSE of various algorithms.

For the S/SB/SPU-LMS algorithms, $A_\psi(\sigma_e^2) = 1$, $B_\psi(\sigma_e^2) = \sigma_e^2$, and $I_i(\Lambda) = I_i'(\Lambda) = I_{ii}(\Lambda) = 1$. Equation (39) will reduce to

$$\begin{aligned}\mathrm{EMSE}_{\mathrm{S_LMS}}(\infty) &\approx \frac{\mu}{2}\sigma_e^2(\infty)\mathrm{Tr}\left(\boldsymbol{U}^T\left[\left(\boldsymbol{U}\Lambda\boldsymbol{U}^T\right)\circ\Omega\right]\boldsymbol{U}\right)\\ &= \frac{\mu}{2}\left(\mathrm{EMSE}_{\mathrm{S_LMS}}(\infty)+\sigma_g^2\right)\mathrm{Tr}\left(\left(\boldsymbol{U}\Lambda\boldsymbol{U}^T\right)\right)\\ &= \frac{\mu}{2}\left(\mathrm{EMSE}_{\mathrm{S_LMS}}(\infty)+\sigma_g^2\right)\mathrm{Tr}(\boldsymbol{R}_{\boldsymbol{XX}}).\end{aligned}$$

Thus, we get

$$\mathrm{EMSE}_{\mathrm{S_LMS}}(\infty) \approx \frac{\frac{1}{2}\mu\sigma_g^2\phi_{\mathrm{LMS}}}{1-\frac{1}{2}\mu\phi_{\mathrm{LMS}}} \quad (40)$$

where $\phi_{\mathrm{LMS}} = \mathrm{Tr}(\boldsymbol{R}_{\boldsymbol{XX}})$. Ignoring the second term in the denominator, (40) will reduce to [14, eq. (31)].

For the S/SB/SPU-LMS algorithms with nonlinearity, since $\sigma_e^2(\infty) = \mathrm{EMSE}(\infty) + \sigma_g^2$, (39) is a nonlinear equation in $\mathrm{EMSE}(\infty)$. In particular, for the S/SB/SPU-LMM algorithms with MH nonlinearity and ATS, $A_{\mathrm{MH}}(\sigma_e^2) \approx \mathrm{erf}(k_\xi/\sqrt{2}) - (2k_\xi/\sqrt{2\pi})\exp(-(k_\xi^2/2)) = A_c$, $B_{\mathrm{MH}}(\sigma_e^2) = \mathrm{erf}(k_\xi\hat{\sigma}_e/\sqrt{2}\sigma_e)\sigma_e^2 - (2\sigma_e\xi/\sqrt{2\pi})\exp(-(k_\xi^2\hat{\sigma}_e^2/2\sigma_e^2)) \approx \mathrm{erf}(k_\xi/\sqrt{2})\sigma_e^2 - (2k_\xi\sigma_e^2/\sqrt{2\pi})\exp(-(k_\xi^2/2)) = \sigma_e^2 A_c$, and $\mathrm{EMSE}(\infty) \approx (\mu B(\sigma_e^2(\infty))/2A_\psi(\sigma_e^2(\infty)))\mathrm{Tr}(\boldsymbol{R}_{\boldsymbol{XX}})$; hence

$$\mathrm{EMSE}_{\mathrm{S_LMM}}(\infty) \approx \frac{\frac{1}{2}\mu\sigma_g^2\phi_{\mathrm{LMS}}}{1-\frac{1}{2}\mu\phi_{\mathrm{LMS}}} \quad (41)$$

which is the same as that of the S-LMS algorithms in (40). In other words, using ATS, the degradation due to the nonlinearity is very small, and the LMS and LMM algorithms will give a similar performance.

For the conventional NLMS/NLMM algorithms, $C = 1$, and $\Omega = \mathbf{1}_L$; we thus obtain

$$\begin{aligned}\mathrm{EMSE}_{\mathrm{NLMS}}(\infty) &\approx \frac{\mu\sigma_e^2(\infty)}{2}\mathrm{Tr}\left(\boldsymbol{D}_\Lambda^{-1}\Lambda\boldsymbol{I}'(\Lambda)\right)\\ &= \frac{\frac{1}{2}\mu\sigma_g^2\phi_{\mathrm{NLMS}}}{1-\frac{1}{2}\mu\phi_{\mathrm{NLMS}}}\end{aligned} \quad (42)$$

where $\phi_{\mathrm{NLMS}} = \mathrm{Tr}(\boldsymbol{D}_\Lambda^{-1}\Lambda\boldsymbol{I}'(\Lambda))$ and

$$\mathrm{EMSE}_{\mathrm{NLMM}}(\infty) \approx \frac{\mu B\left(\sigma_e^2(\infty)\right)}{2A_\psi\left(\sigma_e^2(\infty)\right)}\phi_{\mathrm{NLMS}}. \quad (43)$$

It can be seen that the eigenvalues of $\boldsymbol{R}_{\boldsymbol{XX}}$ are scaled by the matrix $\boldsymbol{D}_\Lambda^{-1}\boldsymbol{I}'(\Lambda)$ as opposed to the conventional LMS algorithm. For the MH nonlinearity, (43) reduces to

$$\mathrm{EMSE}_{\mathrm{NLMM}}(\infty) \approx \frac{\frac{1}{2}\mu\sigma_g^2\phi_{\mathrm{NLMS}}}{1-\frac{1}{2}\mu\phi_{\mathrm{NLMS}}}$$

which is approximately equal to that of its NLMS counterpart. For other values of C, it can be seen from (39) that the rotation $\boldsymbol{U}$ will also affect the steady-state error.

Uncorrelated Inputs: If $x(n)$ is an uncorrelated process such as a white Gaussian input, then $\boldsymbol{R}_{\boldsymbol{XX}} = \Lambda$, and (37) can be simplified to

$$\begin{aligned}\boldsymbol{\Phi}(n+1) \approx \boldsymbol{\Phi}(n) &- \frac{\mu}{C}A_\psi\left(\sigma_e^2\right)\Lambda\boldsymbol{D}_\Lambda\boldsymbol{\Phi}(n)\\ &- \frac{\mu}{C}A_\psi\left(\sigma_e^2\right)\boldsymbol{\Phi}(n)\boldsymbol{D}_\Lambda\Lambda\\ &+ \frac{2\mu^2}{C}C_\psi\left(\sigma_e^2\right)\left\{\left[\Lambda\left(\boldsymbol{\Phi}(n)\circ\boldsymbol{I}(\Lambda)\right)\Lambda\right]\circ\Omega\right\}\\ &+ \frac{\mu^2}{C}B_\psi\left(\sigma_e^2\right)\left[\left(\Lambda\boldsymbol{I}'(\Lambda)\right)\circ\Omega\right].\end{aligned} \quad (44)$$

Its ith diagonal value can be written as follows:

$$\begin{aligned}\Phi_{i,i}(n+1) \approx \Phi_{i,i}(n) &- \frac{2\mu}{C}A_\psi\left(\sigma_e^2\right)I_i(\Lambda)\lambda_i\Phi_{i,i}(n)\\ &+ \frac{2\mu^2}{C}C_\psi\left(\sigma_e^2\right)I_{ii}(\Lambda)\lambda_i^2\Phi_{i,i}(n)\\ &+ \frac{\mu^2}{C}B_\psi\left(\sigma_e^2\right)\lambda_i I_i'(\Lambda).\end{aligned} \quad (45)$$

At the steady state, $n \to \infty$, and (45) can be solved to give $\Phi_{i,i}(\infty) \approx (\mu B_\psi(\sigma_e^2)I_i'(\Lambda)/(2(A_\psi(\sigma_e^2)I_i(\Lambda) - \mu C_\psi(\sigma_e^2)I_{ii}(\Lambda)\lambda_i)))$. Since $\mathrm{EMSE}(\infty) = \Sigma_{i=1}^L\lambda_i\Phi_{i,i}(\infty)$, the steady-state EMSE of the NLMS with nonlinearity and uncorrelated input is

$$\begin{aligned}&\mathrm{EMSE}_{\mathrm{NLMS_\psi_U}}(\infty)\\ &\approx \frac{\mu B_\psi\left(\sigma_e^2(\infty)\right)}{2}\\ &\quad\times\sum_{i=1}^{L}\frac{\lambda_i I_i'(\Lambda)}{A_\psi\left(\sigma_e^2(\infty)\right)I_i(\Lambda) - \mu\lambda_i C_\psi\left(\sigma_e^2(\infty)\right)I_{ii}(\Lambda)}.\end{aligned} \quad (46)$$

Here, the subscripts ψ and U represent the nonlinearity and uncorrelated input, respectively. Since $\mathrm{EMSE}_{\mathrm{S-NLMS_\psi_U}}(\infty) = \sigma_e^2(\infty) - \sigma_g^2$, (46) is a nonlinear equation in $\sigma_e^2(\infty)$. For the S/SB/SPU-NLMS algorithm with

M-nonlinearity and ATS, $A_\psi(\sigma_e^2) \approx A_\psi$, $B_\psi(\sigma_e^2) \approx S_\psi \sigma_e^2$, and $C_\psi(\sigma_e^2) \approx C_\psi$[34]. Then, (46) can be simplified to

$$\mathrm{EMSE}_{\mathrm{NLMM_}\psi\mathrm{_}U}(\infty) \approx \frac{1}{2}\mu\sigma_e^2(\infty)\phi_{\mathrm{NLMM_}\psi} \tag{47}$$

where $\phi_{\mathrm{NLMM_}\psi} = S_\psi \sum_{i=1}^{L}(\lambda_i I_i'(\Lambda) / (A_\psi I_i(\Lambda) - \mu\lambda_i C_\psi I_{ii}(\Lambda)))$ and $S_\psi = B_\psi(\sigma_e^2)/\sigma_e^2$. Using again the fact that $\sigma_e^2(\infty) = \mathrm{EMSE}_{\mathrm{NLMM_}\psi\mathrm{_}U}(\infty) + \sigma_g^2$, (47) can be rearranged to give

$$\mathrm{EMSE}_{\mathrm{NLMM_}\psi\mathrm{_}U}(\infty) \approx \frac{1}{2}\mu\phi_{\mathrm{NLMM_}\psi}\sigma_g^2 / \left(1 - (1/2)\mu\phi_{\mathrm{NLMM_}\psi}\right). \tag{48}$$

Equation (45) is similar to those of the NLMM algorithm in [34], except that A_ψ, S_ψ, and C_ψ are divided by the factor C. Therefore, one can estimate an upper bound of the step size for convergence in the mean square sense by using the result in [34] with A_ψ, S_ψ, and C_ψ being divided by C, and it gives

$$\mu_{B\mathrm{_NLMM_}\psi\mathrm{_}Uq} = \frac{2A_\psi}{S_\psi \Sigma_{i=1}^{L}\lambda_i \left[I_i'(\Lambda)/\left(I_i(\Lambda)\right) + 2\left(\frac{C_\psi}{C}\right) I_{ii}(\Lambda)/I_i(\Lambda)\right]}.$$

We now discuss the relation of the aforementioned result with some known previous results and propose a general guideline for choosing the step size.

Remarks: (R-A3): **LMS algorithm with M-nonlinearity**: When $I_i(\Lambda)$, $I_i'(\Lambda)$, $I_{ii}(\Lambda)$, and C are all equal to one, the analysis will reduce to the LMS algorithm with general M-nonlinearity. Using (37), it can be shown that

$$\mathrm{EMSE}_{\mathrm{LMS_}\psi}(\infty) \approx \frac{\mu B_\psi\left(\sigma_e^2(\infty)\right)}{2} \times \sum_{i=1}^{L} \frac{\lambda_i}{A_\psi\left(\sigma_e^2(\infty)\right) - \mu\lambda_i C_\psi\left(\sigma_e^2(\infty)\right)}. \tag{49}$$

$A_\psi(\sigma_e^2)$, $B_\psi(\sigma_e^2)$, and $C_\psi(\sigma_e^2)$ for some related algorithms are summarized in Table III. These expressions agree with the conventional NLMS algorithms with M-nonlinearity and related algorithms [30], [31], which were studied in more detail in [34].

As mentioned earlier, if $\mu\psi(e(n))$ in (13) is replaced by $\psi_Q(\mu e(n))$, where $\psi_Q(\cdot)$ is a quantizer function, then (13) with $\mu = 1$ (it is absorbed into the quantizer) can be used to model the finite word length effect of the algorithms with $A_\psi(\sigma_e^2)$ and $B_\psi(\sigma_e^2)$ being summarized in Table III. For simplicity, the terms involving $C_\psi(\sigma_e^2)$ are ignored assuming a small step. The usefulness of this model and more simulation results can be found in [36].

(R-A4) **Step size selection for S-LMS family of algorithms**: Using the result in [23], it was shown in [34] that the optimal step size of the LMS algorithm is approximately given by $\mu_{\mathrm{LMS,opt}} = \lambda/((L-1)\lambda^2 + E[x^4])$. Since the maximum possible adaptation speed of the S-LMS algorithm is C times that of the LMS algorithm, we have $\mu_{\mathrm{S-LMS,opt}} = C\mu_{\mathrm{LMS,opt}} \approx C/(\lambda L)$ for a large L. As a result, $\mu_{\mathrm{S-NLMS}} \approx C\alpha$, and one gets the following update with maximum possible step size:

$$\boldsymbol{W}(n+1) = \boldsymbol{W}(n) + \frac{C\boldsymbol{S}_{\boldsymbol{X}}(n)\boldsymbol{X}(n)e(n)}{(\varepsilon/\alpha) + \boldsymbol{X}^T(n)\boldsymbol{X}(n)}.$$

When $C = 1$, it agrees with the optimum data nonlinearity for LMS adaptation in the white Gaussian input obtained in [23] using calculus of variations. The MSE improvement of the NLMS algorithms over the LMS algorithms was analyzed in detail in [23]. In general, one could set $\alpha = 1$ and vary μ between zero and one with a small ε in (4) to achieve a given MSE or to match a given convergence rate, such as the maximum speed previously mentioned.

Similar to the findings in [39] for the NLMS algorithms, we found from the simulation results that the EMSE of the S/SB/SPU-NLMS algorithms varies slightly with the eigenvalues for a given $\mathrm{Tr}(\boldsymbol{R}_{XX})$. For small $\mu_{\mathrm{S-NLMS}}$, (40) and (41) suggest that the S-LMS algorithm is almost independent of the eigenvalue spread for a given $\mathrm{Tr}(\boldsymbol{R}_{XX})(\phi_{\mathrm{S-LMS}} \approx \Sigma_{i=1}^{L}\lambda_i)$. Therefore, the relationship between $\mu_{\mathrm{S-NLMS}}$ and $\mu_{\mathrm{S-LMS}}$ for the white input case, i.e., $\mu_{\mathrm{S-NLMS}} \approx \mu_{\mathrm{S-LMS}}\mathrm{Tr}(\boldsymbol{R}_{XX})$, can be used as a reasonable approximation for the colored case and $\alpha = 1$. The corresponding EMSE is approximately $(1/2)\mu_{\mathrm{S-LMS}}\sigma_g^2\mathrm{Tr}(\boldsymbol{R}_{XX}) = (1/2)\mu_{\mathrm{S-NLMS}}\sigma_g^2$. From the simulation, we also found that the EMSE of the S/SB/SPU-NLMS algorithms will increase slightly with the eigenvalue spread. Hence, $(1/2)\mu_{\mathrm{S-NLMS}}\sigma_g^2$ represents a useful lower bound for estimating the EMSE of the S/SB/SPU-NLMS algorithms. It is attractive because it does not require the knowledge of the eigenvalues or the eigenvalue spread of $\boldsymbol{R}_{XX}$. The corresponding estimate of the misadjustment is then $(1/2)\mu_{\mathrm{S-NLMS}}$.

B. Mean and Mean Square Behaviors in CG Noise

We now briefly analyze the mean and mean square behaviors of the various algorithms in a CG noise environment. Although Price's theorem is originally proposed for Gaussian variates, it was shown later in [40] that it is also applicable to independent mixtures and, hence, Gaussian mixtures. This extension of Price's theorem [21], [22] was employed in the analysis of the LMS and NLMS algorithms with MH nonlinearity and CG noise in [37]. The case of general M-nonlinearity was treated in [34]. Similar techniques were also employed in analyzing the RLM [28] and other related algorithms [33] for the MH nonlinearity.

1) Mean Behavior: Since η_o is now a CG noise as defined in (11), it is a Gaussian mixture consisting of two components $\eta_g(n)$ and $\eta_\Sigma(n)$, each with a zero mean and variances σ_g^2 and σ_Σ^2, respectively. The occurrence probability of the impulsive noise is p_r. Accordingly

$$E_{\{\boldsymbol{v},\boldsymbol{X},\eta_o\}}[f(\boldsymbol{X}(n), e(n))] = (1-p_r)E_{\{\boldsymbol{v},\boldsymbol{X},\eta_g\}}[f(\boldsymbol{X}(n), e(n))] + p_r E_{\{\boldsymbol{v},\boldsymbol{X},\eta_\Sigma\}}[f(\boldsymbol{X}(n), e(n))] \tag{50}$$

where $f(\boldsymbol{X}(n), e(n))$ is an arbitrary quantity whose statistical average is to be evaluated. Since $\boldsymbol{X}(n)$, $\eta_g(n)$, and $\eta_\Sigma(n)$ are Gaussian distributed, each of the expectation on the right-hand

side can be evaluated using Price's theorem. Consequently, the results in Section III-A can be carried forward to the CG noise case by first changing the noise power to σ_g^2 and σ_Σ^2, respectively, and then combining the two results using (50).

Recall the relation of the mean weight-error vector in (14)

$$E\left[\boldsymbol{v}(n+1)\right] = E\left[\boldsymbol{v}(n)\right] - \mu \boldsymbol{H}' \tag{51}$$

where $\boldsymbol{H}' = E_{\{\boldsymbol{v},\boldsymbol{X},\eta_o\}}\left[\psi\left(e(n)\right)\boldsymbol{S}_{\boldsymbol{X}}(n)\,\boldsymbol{X}(n)\,/\,(\varepsilon + \alpha \boldsymbol{X}^T(n)\boldsymbol{X}(n))\right] = (1-p_r)\boldsymbol{H}'_g + p_r \boldsymbol{H}'_\Sigma$ and $\boldsymbol{H}'_g$ and $\boldsymbol{H}'_\Sigma$ are the expectations of the term inside the brackets shown previously with respect to $\{\boldsymbol{v}, \boldsymbol{X}, \eta_g\}$ and $\{\boldsymbol{v}, \boldsymbol{X}, \eta_\Sigma\}$, respectively. From (16) and (17), $\boldsymbol{H}'_i \approx (1/C)\overline{\psi'}(\sigma_{e_i}^2(n))\boldsymbol{U}\Lambda\boldsymbol{D}_\Lambda\boldsymbol{U}^T\boldsymbol{v}(n)$, $i = g, \Sigma$, where $\sigma_{e_g}^2(n) = E[\boldsymbol{v}^T(n)\boldsymbol{R}_{\boldsymbol{XX}}\boldsymbol{v}(n)] + \sigma_g^2$ and $\sigma_{e_\Sigma}^2(n) = E[\boldsymbol{v}^T(n)\boldsymbol{R}_{\boldsymbol{XX}}\boldsymbol{v}(n)] + \sigma_\Sigma^2$. $\boldsymbol{U}$, Λ, and $\boldsymbol{D}_\Lambda$ have been defined in Section III-A. Hence, $\boldsymbol{H}' \approx (1/C)\widetilde{A}_\psi(n)\boldsymbol{U}\Lambda\boldsymbol{D}_\Lambda\boldsymbol{U}^T\boldsymbol{v}(n)$, where $\widetilde{A}_\psi(n) = (1-p_r)\overline{\psi'}(\sigma_{e_g}^2(n)) + p_r\overline{\psi'}(\sigma_{e_\Sigma}^2(n))$. Substituting this into (51) and using the natural coordinate $\boldsymbol{V}(n) = \boldsymbol{U}^T\boldsymbol{v}(n)$, one gets

$$E\left[\boldsymbol{V}(n+1)\right]_i = \left(1 - \frac{\mu}{C}\widetilde{A}_\psi(n)\lambda_i I_i(\Lambda)\right) E\left[\boldsymbol{V}(n)\right]_i. \tag{52}$$

For notational convenience, we have replaced the approximate symbol by the equality symbol. This yields the same form as (21), except for $\widetilde{A}_\psi(n)$. A similar argument regarding the mean convergence in Section III-A also applies to (52). A sufficient condition for the mean weight vector of the algorithm to converge is $|1 - (\mu/C)\widetilde{A}_\psi(n)\lambda_i I_i(\Lambda)| < 1$, for all i's. If $\overline{\psi'}(\sigma_e^2)$ is bounded above by $A_{\psi_\max}$ and so is $\widetilde{A}_\psi(n)$ by $\widetilde{A}_{\psi_\max}$, then, following the argument in Section III-A, the following conservative maximum step size is obtained:

$$\mu_{\max} < 2C/\left(\widetilde{A}_{\psi_\max}\lambda_{\max} I_{i_\lambda_{\max}}(\Lambda)\right). \tag{53}$$

The major convergence results of (53) for the LMS and LMM families of algorithms are summarized by the following remarks.

Remarks: (R-B1): **S-LMS family of algorithms**: In these cases, $\widetilde{A}_\psi(n) = 1$. Compared with the Gaussian case, the convergence rate remains unchanged. All the conclusions in (R-A1) apply.

(R-B2): **S-LMM family of algorithms**: For the M-nonlinearity without ATS, both $\sigma_{e_g}^2$ and $\sigma_{e_\Sigma}^2$ can be very large due to the large value of $\sigma_{e_\Sigma}^2$ and the slow decay of the EMSE $E[\boldsymbol{v}^T(n)\boldsymbol{R}_{\boldsymbol{XX}}\boldsymbol{v}(n)]$, as the gain $\widetilde{A}_\psi(n) = (1-p_r)\overline{\psi'}(\sigma_{e_g}^2(n)) + p_r\overline{\psi'}(\sigma_{e_\Sigma}^2(n))$ can be very small initially. This leads to nonlinear adaptation and slow convergence. Near convergence, $\sigma_e^2(n)$ and $\widetilde{A}_\psi(n)$ will become stable. The convergence is exponential, and the rate for the ith mode is approximately $1 - (\mu/C)\widetilde{A}_\psi(\infty)\lambda_i I_i(\Lambda)$. Normally, the second term $p_r\overline{\psi'}(\sigma_{e_\Sigma}^2(n))$ will be much smaller than the first one due to the clipping property of the nonlinearity. The "asymptotic convergence rate" of the NLMS-based algorithms with M-nonlinearity will still be faster than their LMS counterparts if the eigenvalues are unequal. For the NLMM-based algorithms with ATS, this degradation is not so serious since, if $\sigma_{e_g}^2 \ll \sigma_{e_\Sigma}^2$, $\widetilde{A}_{\mathrm{MH}} \approx (1-p_r)A_c$ is a constant that is close to one if p_r is not too large. Therefore, the fastest convergence rate of these NLMM-based algorithms is still given by (25). Consequently, in additive CG noise, the convergence rate of the NLMM-based algorithms will also be faster than their unnormalized counterparts if the eigenvalues are unequal.

2) Mean Square Behavior: Using a similar approach as in Section III-A2, it can be shown that

$$\begin{aligned}\boldsymbol{\Phi}(n+1) \approx \boldsymbol{\Phi}(n) &- \frac{\mu}{C}\widetilde{A}_\psi(n)\Lambda\boldsymbol{D}_\Lambda\boldsymbol{\Phi}(n) \\ &- \frac{\mu}{C}\tilde{A}_\psi(n)\boldsymbol{\Phi}(n)\boldsymbol{D}_\Lambda\Lambda + \frac{2\mu^2}{C}\widetilde{C}_\psi(n)\boldsymbol{U}^T \\ &\times\left\{\left[\boldsymbol{U}\Lambda\left(\boldsymbol{\Phi}(n)\circ\boldsymbol{I}(\Lambda)\right)\Lambda\boldsymbol{U}^T\right]\circ\Omega\right\}\boldsymbol{U} \\ &+ \frac{\mu^2}{C}\widetilde{B}_\psi(n)\boldsymbol{U}^T\left[\left(\boldsymbol{U}\Lambda\boldsymbol{I}'(\Lambda)\boldsymbol{U}^T\right)\circ\Omega\right]\boldsymbol{U}\end{aligned} \tag{54}$$

where $\widetilde{C}_\psi(n) = (1-p_r)C_\psi(\sigma_{e_g}^2(n)) + p_r C_\psi(\sigma_{e_\Sigma}^2(n))$ and $\widetilde{B}_\psi(n) = (1-p_r)B_\psi(\sigma_{e_g}^2(n)) + p_r B_\psi(\sigma_{e_\Sigma}^2(n))$. Due to page limitation, we only summarize the result for the small step size case

$$\mathrm{EMSE}(n) \approx \mu\frac{\widetilde{B}_\psi(n)}{2\widetilde{A}_\psi(n)}\mathrm{Tr}\left(\boldsymbol{D}_\Lambda^{-1}\boldsymbol{U}^T\left[\left(\boldsymbol{U}\Lambda\boldsymbol{I}'(\Lambda)\boldsymbol{U}^T\right)\circ\Omega\right]\boldsymbol{U}\right). \tag{55}$$

For the S/SB/SPU-LMM algorithms with MH nonlinearity and ATS, $I_i(\Lambda)$, $I'_i(\Lambda)$, and $I_{ii}(\Lambda)$ are all equal to one. Moreover, $\widetilde{A}_{\mathrm{MH}}(n) \approx \widetilde{C}_{\mathrm{MH}}(n) \approx (1-p_r)A_c$ and $\widetilde{B}_{\mathrm{MH}}(n) \approx (1-p_r)\sigma_{e_g}^2(\infty)(\mathrm{erf}(k_\xi/\sqrt{2}) - (2k_\xi/\sqrt{2\pi})\exp(-(k_\xi^2/\sqrt{2}))) = (1-p_r)\sigma_{e_g}^2(\infty)A_c$ as $n \to \infty$. Hence, (55) can be simplified to

$$\mathrm{EMSE}_{\mathrm{S-LMM_SSS_CG}}(\infty) \approx \frac{\frac{1}{2}\mu\sigma_g^2\phi_{\mathrm{LMS}}}{1 - \frac{1}{2}\mu\phi_{\mathrm{LMS}}}. \tag{56}$$

The EMSE is still similar to their conventional LMS-based counterparts in a Gaussian noise environment. This illustrates the robustness of the LMM-based algorithms. For the NLMM-based algorithm with MH nonlinearity, i.e., $C = 1$ and $\Omega = \boldsymbol{1}_L$, we, in turn, obtain

$$\mathrm{EMSE}_{\mathrm{NLMM_SSS_CG}}(\infty) \approx \frac{\frac{1}{2}\mu\sigma_g^2\phi_{\mathrm{NLMS}}}{1 - \frac{1}{2}\mu\phi_{\mathrm{NLMS}}} \tag{57}$$

where $\phi_{\mathrm{NLMS}} = \mathrm{Tr}(\boldsymbol{D}_\Lambda^{-1}\Lambda\boldsymbol{I}'(\Lambda))$. For other values of C, i.e., the PU algorithms, the rotation $\boldsymbol{U}$ in (38) will affect the steady-state error. The EMSE of the conventional LS algorithms, on the other hand, will be considerably increased with the variance and occurrence of the impulsive noises. For instance, in the NLMS algorithm, $\widetilde{A}_\psi = \widetilde{C}_\psi = 1$ and $\widetilde{S}_\psi = (1-p_r)\sigma_g^2 + p_r\sigma_\Sigma^2 = \sigma_g^2 + p_r\sigma_w^2 = \sigma_{\eta_0}^2$; then

$$\mathrm{EMSE}_{\mathrm{NLMS_SSS_CG}}(\infty) \approx \frac{\frac{1}{2}\mu\sigma_{\eta_0}^2\phi_{\mathrm{NLMS}}}{1 - \frac{1}{2}\mu\phi_{\mathrm{NLMS}}}. \tag{58}$$

The increase in the EMSE over the NLMM algorithm is

$$\Delta\mathrm{EMSE}_{\mathrm{NLMS_SSS_CG}}(\infty) \approx \frac{\frac{1}{2}\mu\phi_{\mathrm{NLMS}}}{1 - \frac{1}{2}\mu\phi_{\mathrm{NLMS}}}p_r\sigma_w^2. \tag{59}$$

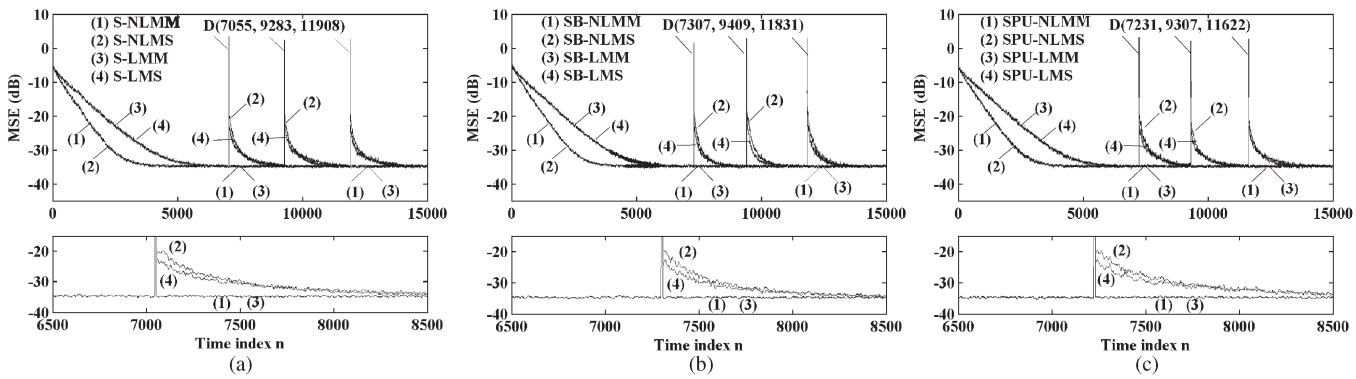


Fig. 3. MSE curves of (a) S-NLMM/NLMS/LMM/LMS, (b) SB-NLMM/NLMS/LMM/LMS, and (c) SPU-NLMM/NLMS/LMM/LMS algorithms with impulses in desired signal, $L = 9$, and $C = 3$.

Similar results are obtained for the LMS algorithm with $I_i(\Lambda) = I_i'(\Lambda) = I_{ii}(\Lambda) = 1$. We note from (59) that the improvement of the M-estimation algorithms over the LS-based algorithm is proportional to the power of the impulsive components, which is reasonable and expected.

IV. Simulation Results

In this section, computer simulations are conducted to evaluate the robustness of the proposed S-LMM family of algorithms to impulsive noises and verify the analytical results obtained in Section III. All simulations were performed using the system identification model shown in Fig. 1, except experiment 3 where we consider the joint ANC and AEC in an automobile with double-talk to illustrate the practical usefulness of the M-estimate algorithms. The unknown system to be estimated in experiments 1 and 2 is an order L finite impulse response filter, and the weight vector $\boldsymbol{W}^*$ is randomly generated and normalized to unit energy. The input signal $x(n)$ is a first-order autoregressive process $x(n) = ax(n-1) + v(n)$, where $v(n)$ is an additive white Gaussian noise sequence with a zero mean and variance σ_v^2. $0 < a < 1$ is the correlation coefficient controlling the degree of correlation among the elements of $x(n)$. This is usually used to model speech signals which are correlated, and the whole setup is similar to that encountered in an AEC. The Gaussian noise and the CG noise are generated from (11) with appropriate parameters. The signal-to-noise ratio (SNR) at the system output is given by $10\log 10(\sigma_{d_0}^2/\sigma_g^2)$, where $\sigma_{d_0}^2$ is the power of the output signal of the unknown system. In the AEC setting, the impulsive noise may arise from the noisy environment such as road traffic or double-talk at both sides. For all the LMM/NLMM-based algorithms, the MH M-estimate function is used. All the learning curves are obtained by averaging the results of 200 independent runs.

Experiment 1—Convergence Performance in Gaussian and CG Noise: The convergence performance of the various algorithms and the robustness of the S/SB/SPU-LMM/NLMM algorithms to impulsive noise are evaluated. The MSE is used as the performance measure. The system order L and decimation factor C are chosen as nine and three, respectively. The correlation coefficient of the input is chosen as $a = 0.7$. The CG noise is generated from (11) with an impulse occurrence probability of $p_r = 0.005$ and an impulsive characteristic ratio of $r_{im} = 300$. The SNR is 35 dB. For illustration purposes, the impulsive noise is applied to the desired signal after time instant $n = 6500$. From $n = 1$ to $n = 6499$, the additive noise is $\eta_g(n)$. To visualize more clearly the impact of the impulsive noise on the desired signal, the locations of the impulses at each experiment are fixed, but their amplitudes are varied according to $\eta_o(n)$. This can be realized by generating a fixed Bernoulli sequence $\overline{b}(n)$ with $p_r = 0.005$ and using it in all the independent runs. The step sizes of the S/SB/SPU-LMM/LMS algorithms are set to 0.01, and those for the S/SB/SPU-NLMM/NLMS algorithms are chosen as 0.1. This enables all the algorithms to reach a similar steady-state MSE. The small positive constant ε used to prevent division by zero for the NLMS/NLMM-based algorithms is set to 0.00001, and α is chosen to be one. The threshold parameters of the M-estimate function in the S/SB/SPU-LMM/NLMM algorithms $\hat{\sigma}_e^2(n)$ and ξ are calculated from (7) and (8), respectively, with a forgetting factor $\lambda_\sigma = 0.99$. The window length N_w is chosen to be nine. The performances of all the tested algorithms are shown in Fig. 3. Note that, since the MSE includes the noise power, large MSEs are observed at the locations of the impulses. At other locations, the MSE will mainly depend on the performance of the algorithms and the additive Gaussian noise component. It can be seen that all the NLMS/NLMM-based algorithms have a faster convergence speed than their LMS/LMM counterparts, and the LMM/NLMM-based algorithms possess almost identical initial convergence performances as their LMS/NLMS counterparts. The S/SB/SPU-LMM/NLMM algorithms are considerably more robust to impulsive noise in the desired signal than their LMS/NLMS counterparts.

Simulations for longer filter lengths and different values of C are performed, and similar results are obtained. Simulations using p_r larger than 0.005 are also conducted, and in general, the performance of the algorithms will degrade gradually as p_r increases. More simulation results concerning the effects of using different parameter values of step size, SNR, N_w, and k_ξ are available in [33]. The results show that the S-LMM family of algorithms have an improved robustness to impulsive noise and are not too sensitive to these parameters once they are reasonably chosen as suggested.

Experiment 2—Verification of Analytical Results: Computer simulations were performed to verify the theoretical analysis

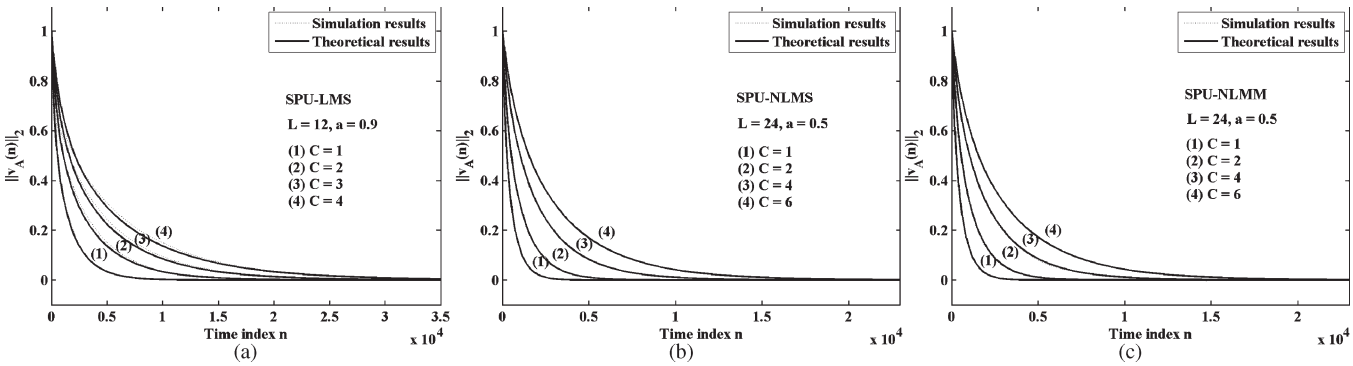


Fig. 4. Mean convergence performances of the (a) SPU-LMS algorithm in Gaussian noise with $L = 12$, $a = 0.9$, and $\mu_{\mathrm{SPU-LMS}} = 0.016$, (b) SPU-NLMS algorithm in Gaussian noise with $L = 24$, $a = 0.5$, and $\mu_{\mathrm{SPU-NLMS}} = 0.1$, and (c) SPU-NLMM algorithm in CG noise with $L = 24$, $a = 0.5$, and $\mu_{\mathrm{SPU-NLMM}} = 0.1$.

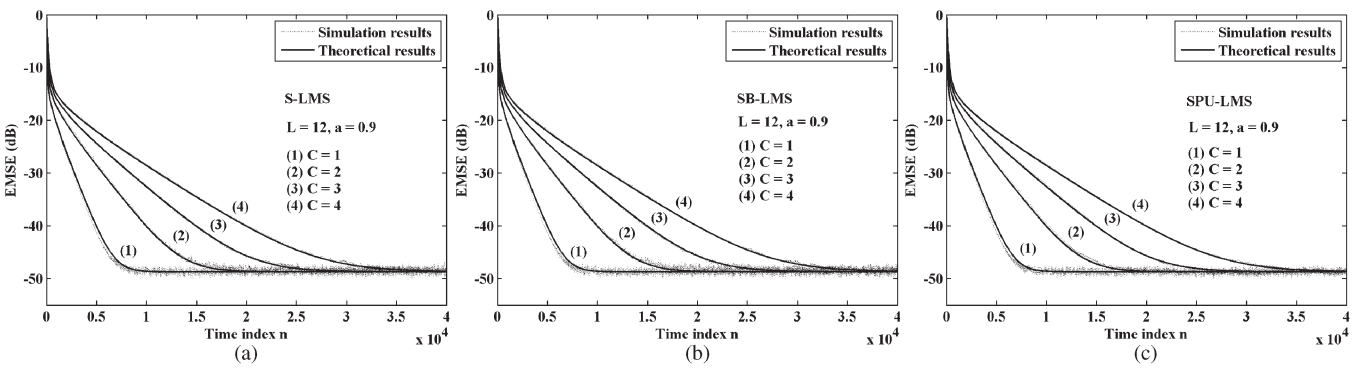


Fig. 5. Mean square convergence performances of the (a) S-LMS, (b) SB-LMS, and (c) SPU-LMS algorithms in Gaussian noise with $L = 12$, $a = 0.9$, and $\mu_{\mathrm{S-LMS}} = \mu_{\mathrm{SB-LMS}} = \mu_{\mathrm{SPU-LMS}} = 0.01$.

presented in Section III. Simulation results for Gaussian noise (see Section III-A) and CG noise (see Section III-B) will be presented. For the latter, the impulsive noise is applied to the desired signals for the tested algorithms throughout the whole adaptation process. Their locations are not fixed and changed according to $\eta_o(n)$ for each independent run. The system order L and correlation coefficient a for the input signal are chosen as $L = 12$ and $a = 0.9$, respectively. To illustrate the effect of decimation factor C, we choose $C = 1$, 2, 3, and 4 for all algorithms. The values of the special integral functions $I_i(\Lambda)$, $I_{ij}(\Lambda)$, and $I'_i(\Lambda)$ defined in (18), (33), and (34), respectively, are evaluated numerically using the method introduced in [41]. The step sizes used and other remaining simulation settings are the same as those in experiment 1.

For the mean convergence, the norm of the mean square weight-error vector is used as the performance measure

$$\|\boldsymbol{v}_A(n)\|_2 = \sqrt{\Sigma_{i=1}^{L}\left[\frac{1}{K}\Sigma_{j=1}^{K} v_i^{(j)}(n)\right]^2}, \qquad i = 1, \ldots, L,; \quad j = 1, \ldots, K$$

where $v_i^{(j)}(n)$ is the ith component of the weight-error vector $\boldsymbol{v}(n)$ at time n in the jth independent run. K is the total number of independent runs, which is set to 200 in this experiment. The theoretical results for the S-LMS family of algorithms in Gaussian noise are computed from (21). Since the results of the S-LMM family are also similar, they are omitted here. For the CG noise case, we only plot the theoretical results for the S-LMM family of algorithms from (52) because the results are similar. To save space, Fig. 4 only shows the mean convergence performance of the SPU-based algorithms under selected experimental settings. A good agreement between the theoretical and simulation results can be observed. Similar results can be obtained for the S- and SB-based algorithms. Interested readers are referred to [36] for more details.

For the mean square convergence of all the algorithms, $\mathrm{EMSE}(n) = \mathrm{Tr}(\boldsymbol{\Phi}(n)\Lambda)$ is used as the performance measure. For the Gaussian noise case, the theoretical results of the S-LMS family of algorithms are computed from (37), and for the CG noise, those of the S-LMM family of algorithms are computed from (54). Similarly, the experimental results are shown in Figs. 5–8. We can see that the convergence speed of all algorithms decreases as C increases. The theoretical and simulation results are also in good agreement with each other.

Due to page limitation, the simulation results on the effect of step size μ, input covariance matrix, σ_g^2, p_r, and r_{im} for the impulsive model in (11) are not shown here. Only the effect of the decimation factor C, which is a key parameter of PU algorithms for trading performance and computational complexity, is shown. Moreover, the simulation results

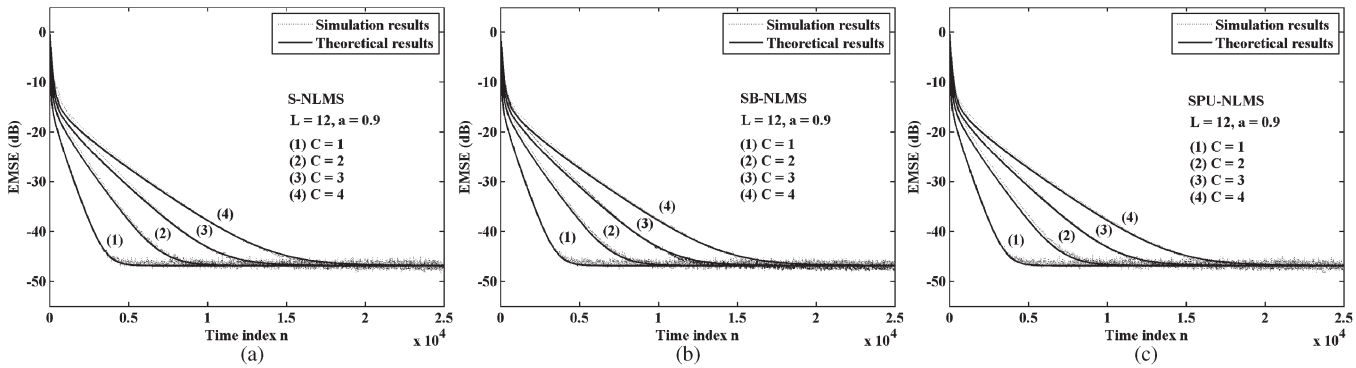


Fig. 6. Mean square convergence performances of the (a) S-NLMS, (b) SB-NLMS, and (c) SPU-NLMS algorithms in Gaussian noise with $L = 12$, $a = 0.9$, and $\mu_{\mathrm{S-NLMS}} = \mu_{\mathrm{SB-NLMS}} = \mu_{\mathrm{SPU-NLMS}} = 0.1$.

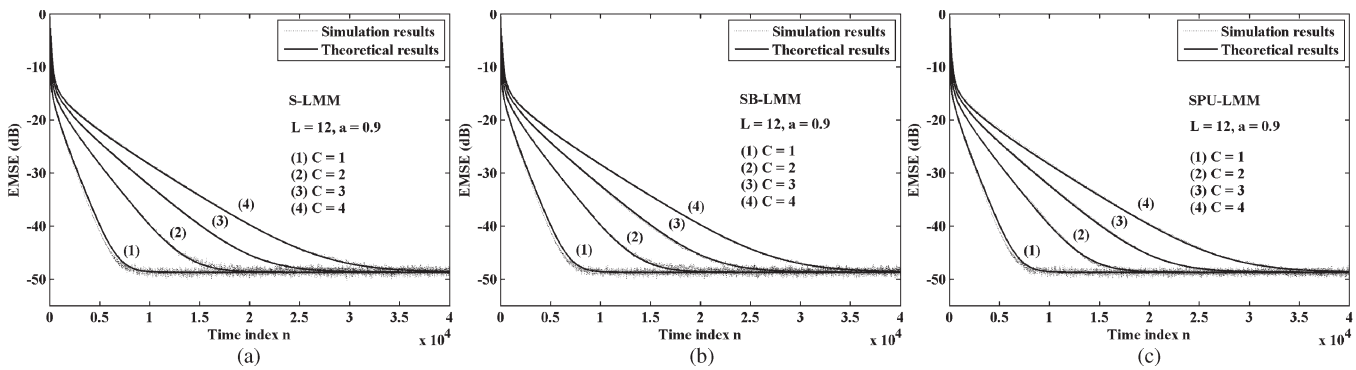


Fig. 7. Mean square convergence performances of the (a) S-LMM, (b) SB-LMM, and (c) SPU-LMM algorithms in CG noise with $L = 12$, $a = 0.9$, and $\mu_{\mathrm{S-LMM}} = \mu_{\mathrm{SB-LMM}} = \mu_{\mathrm{SPU-LMM}} = 0.01$.

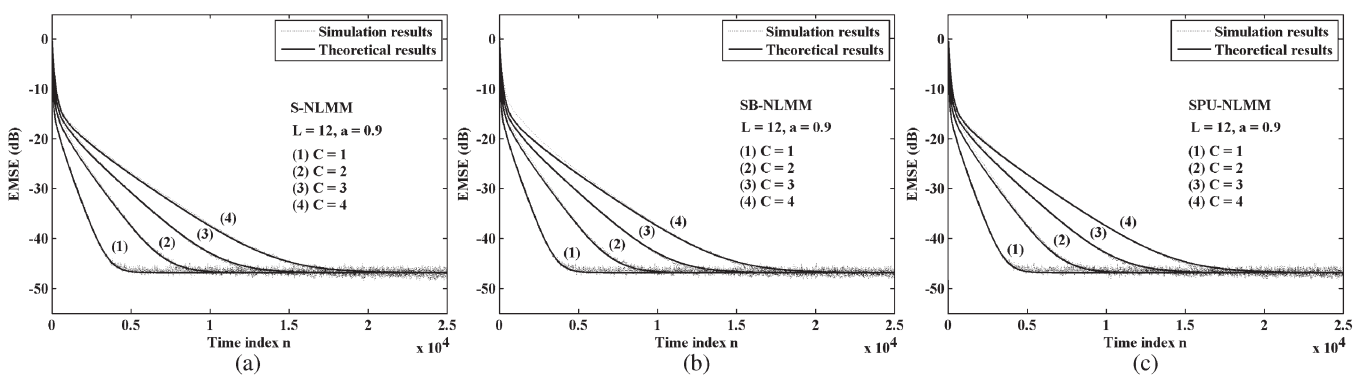


Fig. 8. Mean square convergence performances of the (a) S-NLMM, (b) SB-NLMM, and (c) SPU-NLMM algorithms in CG noise with $L = 12$, $a = 0.9$, and $\mu_{\mathrm{S-NLMM}} = \mu_{\mathrm{SB-NLMM}} = \mu_{\mathrm{SPU-NLMM}} = 0.1$.

for $L = 24$ and $a = 0.5$ can be found in the supplementary materials [36]. As for the effect of the other parameters on the conventional LMS/SNLMS and LMM/NLMM algorithms in Gaussian and CG noise, interested readers are referred to [34] for more details. It was found in [34] that, in CG noise, the MSE performances of the LMM/NLMM-based algorithms are considerably better than their LMS/NLMS-based counterparts. Like the analytical results obtained in this paper, the detailed convergence speed and steady-state EMSE of the LMS/LMM/NLMS/NLMM algorithms depend on the step size, covariance of the input, σ_g^2, and p_r and r_{im} for the CG noise. The simulation results show that similar arguments as in [34] apply to the S/SB/SPU-based algorithms due to their close relationship with the LMS/LMM/NLMS/NLMM algorithms.

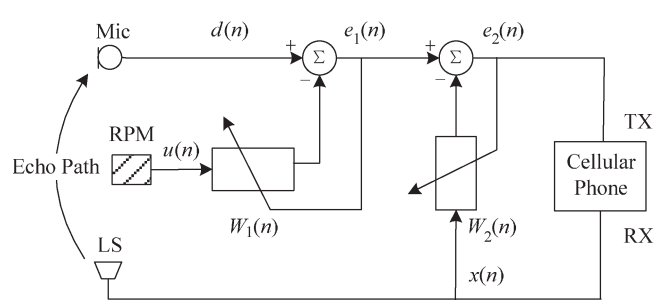


Fig. 9. Hands-free cellular phone model.

Overall, we can see that the theoretical analysis is accurate, and it can provide accurate prediction of the EMSE and a guideline for choosing the step size as suggested in (R-A4).

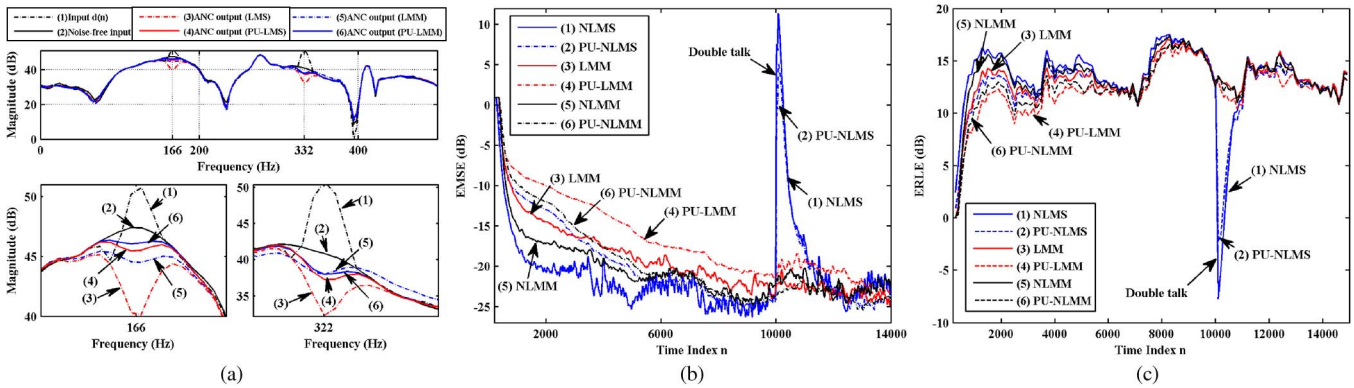


Fig. 10. Design results in experiment 3. (a) Spectra of the desired signal $d(n)$ and the output of ANC $e_1(n)$. (b) Convergence curve of $\|\boldsymbol{v}\|_2$ for various algorithms in AEC: (1) NLMS, (2) PU-NLMS, (3) LMM, (4) PU-LMM, (5) NLMM, and (6) PU-NLMM. (c) Convergence curve of ERLE for various algorithms in AEC: (1) NLMS, (2) PU-NLMS, (3) LMM, (4) PU-LMM, (5) NLMM, and (6) PU-NLMM. For clarity, all ERLE curves are plotted for every 100 time samples.

Experiment 3—Joint ANC and AEC in Automobiles: In this experiment, the PU-LMM and PU-NLMM algorithms are applied to an integrated acoustic signal processing system, which performs ANC and AEC for a hands-free cellular phone inside an automobile [43]. As shown in Fig. 9, the system has three inputs. They are the revolutions-per-minute (RPM) signal from the car's engine, the received far-end signal from the cellular phone (RX), and the near-end signal from the microphone (Mic). The microphone is assumed to be placed at the headrest of the driver's seat, and it is used to pick up the near-end speech signal. The ANC aims to cancel out the engine noise recorded by the microphone by adaptively estimating this noise and subtracting it from the microphone input signal. A speed or rotation sensor is used to measure the RPM of the engine and generate the RPM signal. This is then used to generate a reference signal $u(n)$, which consists of unit magnitude sinusoids at the fundamental rotational frequency of the engine and a certain number of its harmonics. The ANC, which is implemented as an adaptive linear transversal filter, adjusts continuously the complex amplitudes of the reference signal input $u(n)$ to minimize the difference between its output, which is the estimated engine noise, and that of the microphone input. In doing so, the engine noise in the microphone can be suppressed before transmitting to the far end (TX) through the cellular phone. Mathematically, the error signal $e_1(n)$ to be minimized is $e_1(n) = d(n) - \boldsymbol{W}_1^T(n)\boldsymbol{u}(n)$, where $d(n)$ is the desired signal received at the microphone, $\boldsymbol{W}_1(n) = [w_1^{(0)}(n) \quad \cdots \quad w_1^{(N-1)}(n)]^T$ is the weight vector at time instant n of the adaptive-filter-based ANC of length N, and $\boldsymbol{u}(n) = [u(n) \quad \cdots \quad u(n-N+1)]^T$ is the input signal vector containing the reference input at time instant n. Conventionally, the adaptive-filter-based ANC is updated by the LMS or NLMS algorithms. However, impulsive interference may appear at the desired input during double-talk, i.e., when the speaker in the automobile tries to talk when the far-end speaker is talking, or noisy traffic conditions. This may significantly affect the performance of the ANC, as we shall illustrate by the simulation results hereinafter. Therefore, we propose to update the weight vector of the adaptive filter (ANC) by the proposed PU-LMM algorithm, which is able to suppress the adverse effect of the impulsive noise. For the multichannel ANC, a filtered-s LMS algorithm and its variants have been proposed to address the nonlinearities encountered in ANC systems [46]. This is different from the nonlinearity introduced in this paper which is mainly used to combat the adverse effect of impulsive noise.

To suppress the echo from the loudspeaker (see "LS" in Fig. 9) to the microphone, the output of the ANC is forwarded to an AEC, which identifies the acoustic path from the loudspeaker to the microphone by using an adaptive filter $\boldsymbol{W}_2$ with the far-end received signal $x(n)$ and the noise-suppressed microphone input $e_1(n)$ as the signal input and desired input, respectively. The AEC system output $e_2(n)$ is obtained by subtracting the ANC output from the output of the adaptive filter $\boldsymbol{W}_2^T(n)\boldsymbol{x}(n)$, which is the estimated echo component, and $\boldsymbol{W}_2(n) = [w_2^{(0)}(n) \quad \cdots \quad w_2^{(M-1)}(n)]^T$ is the estimated impulse response of the echo path. Hence, $e_2(n) = e_1(n) - \boldsymbol{W}_2^T(n)\boldsymbol{x}(n)$, where $\boldsymbol{x}(n) = [x(n) \quad \cdots \quad x(n-M+1)]^T$ is the input signal vector and M is the order of the filter. To suppress the adverse effect of possible impulsive noise at the desired input, i.e., the ANC output, the weight vector is again updated by the proposed PU-LMM/NLMM algorithms.

In the simulation, the engine is assumed to be running at an idling RPM of 4980, and the noise is assumed to contain sinusoids with frequencies of 166 and 332 Hz. Following the descriptions in [43], we assume that the sinusoid noise is given by $0.0707[\cos(2\pi \cdot 166 \cdot nT) + \cos(2\pi \cdot 332 \cdot nT)]$, where T is the sampling period, and it propagates to the microphone with an amplitude decay of 0.5. The far-end signal is a segment of the speech signal. The double-talk is assumed to occur at the 10 000th sample, and it is stimulated by injecting a short segment of the speech signal of about 100 samples with a large magnitude at the near-end microphone. Both far- and near-end signals are sampled at 8 kHz. The SNR is set to be 15 dB. For both LMS and LMM algorithms, the order of the adaptive filter $\boldsymbol{W}_1$ is 256, and the step size is 0.015, which results in a bandwidth of about 20 Hz [43]. The decimation factor for the PU algorithms is 16, and hence, their arithmetic complexities are about 1/16 of their full-update counterparts. The orders of the echo path and the adaptive filter $\boldsymbol{W}_2$ are both chosen to be 18, and the decimation factor for the PU algorithms is three. To achieve a similar steady-state EMSE, the step sizes for the LMS/LMM and the NLMS/NLMM algorithms are chosen as 0.02 and 0.1, respectively.

Fig. 10(a) shows the spectra of the system input and outputs by the LMS/LMM/PU-LMS/PU-LMM algorithms for the samples in the interval from 9500 to 10 523 using a 1024-point discrete Fourier transform with a Hamming window. It can be seen that the LMM PU-LMS/LMM algorithms offer similar spectra as the noise-free input spectrum while the LMS algorithm introduces large distortions at around 166 and 332 Hz due to the double-talk problem. Since the double-talk signal is relatively short and has a large magnitude, it has considerable components at the fundamental and second harmonics, i.e., 166 and 332 Hz. The LMS adaptive filter has mistaken them as a sharp change in the engine noise and hence attenuates the signal energy at around 166 and 332 Hz. The PU-LMS algorithm is less sensitive to the impulsive noise possibly because the weight vector is updated partially. On the other hand, the LMM/PU-LMM algorithms are much less affected by the double-talk with a sufficiently long median filter length N_w, for example, 200 (the parameters κ, λ_σ, and c_1 are set as the same as in the aforementioned experiments), which is able to combat the adverse effect of the double-talk. After the double-talk, $\boldsymbol{W}_1$ for the LMS and PU-LMS will converge to the desired value again. Given a sufficient long period of time, the spectra obtained by various algorithms are very similar, and therefore, they are not shown here to save space. Since the input $u(n)$ has constant input signal power, the performance of the LMS (LMM) algorithm is the same as that of the NLMS (NLMM) algorithm for the same EMSE. Therefore, only the former results are shown.

Now, we study the performance of the AEC using various algorithms. For a fair comparison, the ANC is performed using the PU-LMM algorithm. Therefore, the input to the AEC e_1 is the same for all the testing algorithms. Fig. 10(b) shows the convergence curves of the EMSE for the algorithms tested since this is related to the amount of the uncanceled echo. The step sizes of the LMM, NLMS, and NLMM algorithms are set to 0.013, 0.1, and 0.1, respectively, so that the steady-state EMSEs of the algorithms are approximately the same. Since the speech signal contains momentarily silence between phonemes, the parameter ε in the NLMS and NLMM, which is also a regularization parameter, is set to 0.3 to avoid the problem of no excitation.

The NLMS algorithms are seen to be significantly affected by the double-talk, while the LMM/NLMM algorithms are rather insensitive to the double-talk. The LMM algorithm is slightly better than the PU-LMM algorithm, which, however, has a considerably lower arithmetic complexity. The LMS and PU-LMS algorithms have a similar performance as the LMM-based algorithms but are considerably more sensitive to the double-talk. Therefore, the results are not shown for space limitation. To further evaluate the performance of the AEC, the echo return loss enhancement (ERLE) defined as $\mathrm{ERLE}(n) = 10\log(E[e_1^2(n)]/E[e_2^2(n)])$ for various algorithms is shown in Fig. 10(c). Note that, for clarity, the ERLE curves in Fig. 10(c) are plotted for every 100 time samples. It can be seen that the performances of the M-estimation-based algorithms are much better than their LS counterparts in the presence of double-talk. On the other hand, the initial convergences and EMSE of the M-estimation algorithms are slightly impaired since some of the error samples are suppressed due to the use of a relatively large median filter length. The lower complexity PU algorithms are seen to have slightly lower performances as compared with their full-update versions.

V. Conclusion

New extensions of conventional S-LMS algorithms and their convergence behaviors with Gaussian inputs and additive Gaussian or CG noises have been presented. The S-LMM family of algorithms proposed are nonlinear extensions of the S-LMS algorithms. They are shown to offer improved performances in adaptive system identification over their conventional LMS/NLMS-based counterparts in an impulsive noise environment. Difference equations describing their mean and mean square convergence behaviors in Gaussian inputs and Gaussian or CG noises are derived. The analytical results reveal the advantages of the normalization in convergence speed and the improved robustness of the S-LMM family in an impulsive noise environment. The theoretical results are in good agreement with the computer simulations. Finally, a simulated example of a joint ANC and AEC for communications inside an automobile was conducted to illustrate the effectiveness of the proposed algorithm in the presence of double-talk.

References

[1] L. Ljung, *System Identification: Theory for the User*. Englewood Cliffs, NJ: Prentice-Hall, 1987.

[2] X. M. Jian, N. Guo, M. X. Li, and H. L. Zhang, "Characterization of bonding quality in a multilayer structure using segment adaptive filtering," *J. Nondestruct. Eval.*, vol. 21, no. 2, pp. 55–65, Jun. 2002.

[3] M. Karimi-Ghartemani and M. R. Iravani, "A nonlinear adaptive filter for online signal analysis in power systems: Applications," *IEEE Trans. Power Del.*, vol. 17, no. 2, pp. 617–622, Apr. 2002.

[4] N. V. Thakor and Y. S. Zhu, "Applications of adaptive filtering to ECG analysis: Noise cancellation and arrhythmia detection," *IEEE Trans. Biomed. Eng.*, vol. 38, no. 8, pp. 785–794, Aug. 1991.

[5] N. Kalouptsidis and S. Theodoridis, *Adaptive System Identification and Signal Processing Algorithms*. Englewood Cliffs, NJ: Prentice-Hall, 1993.

[6] B. Widrow, J. McCool, M. G. Larimore, and C. R. Johnson, Jr., "Stationary and nonstationary learning characteristics of the LMS adaptive filter," *Proc. IEEE*, vol. 64, no. 8, pp. 1151–1162, Aug. 1976.

[7] J. I. Nagumo and A. Noda, "A learning method for system identification," *IEEE Trans. Autom. Control*, vol. AC-12, no. 3, pp. 282–287, Jun. 1967.

[8] K. C. Ho and S. D. Blunt, "Rapid identification of a sparse impulse response using an adaptive algorithm in the Haar domain," *IEEE Trans. Signal Process.*, vol. 51, no. 3, pp. 628–638, Mar. 2003.

[9] 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.

[10] K. Dogancay and O. Tanrikulu, "Adaptive filtering algorithms with selective partial updates," *IEEE Trans. Circuits Syst. II, Analog Digit. Signal Process.*, vol. 48, no. 8, pp. 762–769, Aug. 2001.

[11] A. Tandon, M. N. S. Swamy, and M. O. Ahmad, "Partial-update L_∞-norm based algorithms," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 54, no. 2, pp. 411–419, Feb. 2007.

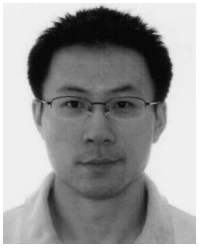
[12] S. Werner, M. L. R. Campos, and P. S. R. Diniz, "Partial-update NLMS algorithms with data-selective updating," *IEEE Trans. Signal Process.*, vol. 52, no. 4, pp. 938–949, Apr. 2004.

[13] M. Godavarti and A. O. Hero, III, "Partial update LMS algorithms," *IEEE Trans. Signal Process.*, vol. 53, no. 7, pp. 2382–2399, Jul. 2005.

[14] S. C. Douglas, "Adaptive filters employing partial updates," *IEEE Trans. Circuits Syst. II, Analog Digit. Signal Process.*, vol. 44, no. 3, pp. 209–216, Mar. 1997.

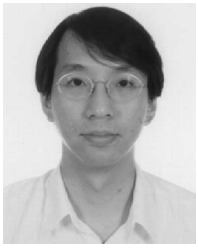
[15] P. Ramos, R. Torrubia, A. López, A. Salinas, and E. Masgrau, "Step size bound of the sequential partial update LMS algorithm with periodic input signals," *EURASIP J. Audio, Speech, Music Process.*, vol. 2007, no. 1, p. 7, Jan. 2007.

[16] S. M. Kuo and J. Chen, "Multiple-microphone acoustic echo cancellation system with the partial adaptive process," *Digit. Signal Process.*, vol. 3, no. 1, pp. 54–63, Jan. 1993.
[17] T. Schertler, "Selective block update of NLMS type algorithms," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, May 1998, vol. 3, pp. 1717–1720.
[18] S. C. Chan, Y. Zhou, and K. L. Ho, "A new sequential block partial update normalized least mean M-estimate algorithm and its convergence performance analysis," in *Proc. IEEE Int. Symp. Signal Process. Inf. Technol.*, 2007, pp. 327–332.
[19] P. J. Huber, *Robust Statistics*. New York: Wiley, 1981.
[20] Y. Zou, S. C. Chan, and T. S. Ng, "Least mean M-estimate algorithms for robust adaptive filtering in impulsive noise," *IEEE Trans. Circuits Syst. II, Analog Digit. Signal Process.*, vol. 47, no. 12, pp. 1564–1569, Dec. 2000.
[21] R. Price, "A useful theorem for nonlinear devices having Gaussian inputs," *IEEE Trans. Inf. Theory*, vol. IT-4, no. 2, pp. 69–72, Jun. 1958.
[22] A. Papoulis, *Probability, Random Variables, and Stochastic Processes,* 3rd ed. New York: McGraw-Hill, 1991.
[23] S. C. Douglas and T. H. Y. Meng, "Normalized data nonlinearities for LMS adaptation," *IEEE Trans. Signal Process.*, vol. 42, no. 6, pp. 1352–1365, Jun. 1994.
[24] T. I. Haweel and P. M. Clarkson, "A class of order statistic LMS algorithms," *IEEE Trans. Signal Process.*, vol. 40, no. 1, pp. 44–53, Jan. 1992.
[25] R. Settineri, M. Najim, and D. Ottaviani, "Order statistic fast Kalman filter," in *Proc. IEEE Int. Symp. Circuits Syst.*, 1996, vol. 2, pp. 116–119.
[26] S. Koike, "Adaptive threshold nonlinear algorithm for adaptive filters with robustness against impulsive noise," *IEEE Trans. Signal Process.*, vol. 45, no. 9, pp. 2391–2395, Sep. 1997.
[27] J. F. Weng and S. H. Leung, "Adaptive nonlinear RLS algorithm for robust filtering in impulsive noise," in *Proc. IEEE Int. Symp. Circuits Syst.*, Jun. 1997, vol. 4, pp. 2337–2340.
[28] S. C. Chan and Y. Zou, "A recursive least M-estimate algorithm for robust adaptive filtering in impulsive noise: Fast algorithm and convergence performance analysis," *IEEE Trans. Signal Process.*, vol. 52, no. 4, pp. 975–991, Apr. 2004.
[29] F. R. Hampel, E. M. Ronchetti, P. J. Rousseeuw, and W. A. Stahel, *Robust Statistics: The Approach Based on Influence Functions*. New York: Wiley, 2005.
[30] N. J. Bershad, "On error-saturation nonlinearities in LMS adaptation," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. 36, no. 4, pp. 440–452, Apr. 1988.
[31] V. J. Mathews, "Performance analysis of adaptive filters equipped with the dual sign algorithm," *IEEE Trans. Signal Process.*, vol. 39, no. 1, pp. 85–91, Jan. 1991.
[32] S. Koike, "Performance analysis of the normalized LMS algorithm for complex-domain adaptive filters in the presence of impulse noise at filter input," *IEICE Trans. Fundam. Electron., Commun. Comput. Sci.*, vol. E89-A, no. 9, pp. 2422–2428, Sep. 2006.
[33] Y. Zhou, "Improved analysis and design of efficient adaptive transversal filtering algorithms with particular emphasis on noise, input and channel modeling," Ph.D. dissertation, Univ. Hong Kong, Hong Kong, 2006.
[34] S. C. Chan and Y. Zhou, "On the performance analysis of the least mean M-estimate and normalized least mean M-estimate algorithms with Gaussian inputs and additive Gaussian and contaminated Gaussian noises," *J. Signal Process. Syst.*, vol. 60, no. 1, pp. 81–103, Jul. 2009.
[35] J. W. Tukey, *A Survey of Sampling From Contaminated Distributions in Contributions to Probability and Statistics*, I. Olkin, Ed. Stanford, CA: Univ. Press, 1960.
[36] S. C. Chan, Y. Zhou, and K. L. Ho, *Supplementary document: New partial update least mean M-estimate algorithms for robust adaptive system identification in impulsive noise*, Digit. Signal Process. Lab., Dept. EEE, Univ. Hong Kong, Hong Kong. [Online]. Available: www.eee.hku.hk/~dsp/TIE2010_supp_document.pdf
[37] S. C. Chan and Y. Zhou, "On the convergence analysis of the normalized LMS and the normalized least mean M-estimate algorithms," in *Proc. IEEE Int. Symp. Signal Process. Inf. Technol.*, 2007, pp. 1048–1053.
[38] J. C. M. Bermudez and N. J. Bershad, "A nonlinear analytical model for the quantized LMS algorithm—The arbitrary step size case," *IEEE Trans. Signal Process.*, vol. 44, no. 5, pp. 1175–1183, May 1996.
[39] S. C. Chan and Y. Zhou, "Convergence behavior of NLMS algorithm for Gaussian inputs: Solutions using generalized Abelian integral functions and step size selection," *J. Signal Process. Syst.*, vol. 59, no. 3, pp. 255–265, Jun. 2010.
[40] R. Price, "Comment on: 'A useful theorem for nonlinear devices having Gaussian inputs'," *IEEE Trans. Inf. Theory*, vol. IT-10, no. 2, p. 171, Apr. 1964.
[41] G. Recktenwald, *Numerical Methods With MATLAB: Implementations and Applications*. Englewood Cliffs, NJ: Prentice-Hall, 2000.
[42] N. J. Bershad, "Analysis of the normalized LMS algorithm with Gaussian inputs," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. ASSP-34, no. 4, pp. 793–806, Apr. 1986.
[43] S. M. Kuo, H. Chuang, and P. P. Mallela, "Integrated automotive signal processing and audio system," *IEEE Trans. Consum. Electron.*, vol. 39, no. 3, pp. 522–532, Aug. 1993.
[44] A. Rosado-Munoz, M. Bataller-Mompean, E. Soria-Olivas, C. Scarante, and J. Guerrero-Martinez, "FPGA implementation of an adaptive filter robust to impulsive noise: Two approaches," *IEEE Trans. Ind. Electron.*, vol. 58, no. 3, pp. 860–870, Mar. 2011.
[45] H. Dai and N. K. Sinha, "A robust off-line output error method for system identification," *IEEE Trans. Ind. Electron.*, vol. 39, no. 4, pp. 285–292, Aug. 1992.
[46] P. V. Kumar, K. M. M. Prabhu, and D. P. Das, "Block filter-s least mean square for active control of non-linear noise system," *IET Signal Process.*, vol. 4, no. 2, pp. 168–180, Apr. 2010.



Y. Zhou (S'05–M'07) received the B.Sc. and M.Sc. degrees in control engineering from the Harbin Institute of Technology, Harbin, China, in 1997 and 1999, respectively, and the Ph.D. degree in electrical engineering from The University of Hong Kong, Pokfulam, Hong Kong, in 2006.

From July 2006 to May 2009, he was a Postdoctoral Research Fellow with the Department of Electrical and Electronic Engineering, The University of Hong Kong. Since August 2009, he has been with the Key Laboratory of Noise and Vibration Research, Institute of Acoustics, Chinese Academy of Sciences, Beijing, China, where he is currently an Associate Professor. His research interests include adaptive signal processing, statistical signal processing, system identification, and estimation theory. His current research focuses on statistical signal processing theory and techniques applied to acoustics, e.g., acoustic echo cancellation, noise reduction, and speech signal processing.



S. C. Chan (S'87–M'92) received the B.Sc. (Eng) and Ph.D. degrees from The University of Hong Kong, Pokfulam, Hong Kong, in 1986 and 1992, respectively.

In 1990, he was with the City Polytechnic of Hong Kong, Kowloon, Hong Kong, where he was an Assistant Lecturer and, later, a University Lecturer. Since 1994, he has been with the Department of Electrical and Electronic Engineering, The University of Hong Kong, where he is currently a Professor. He was a Visiting Researcher with Microsoft Corporation, Redmond, WA, Microsoft, Beijing, China, University of Texas at Arlington, Arlington, and Nanyang Technological University, Singapore. His research interests include fast transform algorithms, filter design and realization, multirate and biomedical signal processing, communications and array signal processing, high-speed A/D converter architecture, bioinformatics, and image-based rendering.

Dr. Chan is currently a member of the Digital Signal Processing Technical Committee of the IEEE Circuits and Systems Society and an Associate Editor of the *Journal of Signal Processing Systems*. He was the Chairman of the IEEE Hong Kong Chapter of Signal Processing in 2000–2002 an organizing committee member of the IEEE International Conference on Acoustics, Speech, and Signal Processing 2003 and the International Conference on Image Processing 2010, and an Associate Editor of the IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS I in 2008–2009.

K. L. Ho (M'78–SM'04) received the B.Sc. (Eng.) and the M.Phil. degrees in electrical engineering from The University of Hong Kong, Pokfulam, Hong Kong, in 1971 and 1973, respectively, and the Ph.D. degree from the University of London, London, U.K., in 1977.

Since 1984, he has been with the Department of Electrical and Electronic Engineering, The University of Hong Kong. His current research interests include signal processing and communications systems.

Dr. Ho is a member of The Hong Kong Institution of Engineers.