

# 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

S. C. Chan · Y. Zhou

Received: 7 March 2009 / Revised: 24 August 2009 / Accepted: 24 August 2009 / Published online: 17 October 2009  
© 2009 Springer Science + Business Media, LLC. Manufactured in The United States

**Abstract** This paper studies the convergence analysis of the least mean M-estimate (LMM) and normalized least mean M-estimate (NLMM) algorithms with Gaussian inputs and additive Gaussian and contaminated Gaussian noises. These algorithms are based on the M-estimate cost function and employ error nonlinearity to achieve improved robustness in impulsive noise environment over their conventional LMS and NLMS counterparts. Using the Price's theorem and an extension of the method proposed in Bershad (*IEEE Transactions on Acoustics, Speech, and Signal Processing*, ASSP-34(4), 793–806, 1986; *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 35(5), 636–644, 1987), we first derive new expressions of the decoupled difference equations which describe the mean and mean square convergence behaviors of these algorithms for Gaussian inputs and additive Gaussian noise. These new expressions, which are expressed in terms of the generalized Abelian integral functions, closely resemble those for the LMS algorithm and allow us to interpret the convergence performance and determine the step size stability bound of the studied algorithms. Next, using an extension of the Price's theorem for Gaussian mixture, similar results are obtained for additive contaminated Gaussian noise case. The theoretical analysis and the

practical advantages of the LMM/NLMM algorithms are verified through computer simulations.

**Keywords** Adaptive filtering · Robust statistics · Least mean square/M-estimate · Impulsive noise

## 1 Introduction

Adaptive filters are frequently employed to handle filtering problems in which the statistics of the underlying signals are either unknown *a priori*, or in some cases, slowly-varying. Many adaptive filtering algorithms have been proposed and they are usually variants of the well known least mean square (LMS) [1] and the recursive least squares (RLS) [2] algorithms. An important variant of the LMS algorithm is the normalized least mean square (NLMS) algorithm [3, 4], where the step size is normalized with respect to the energy of the input vector. Due to the numerical stability and computational simplicity of the LMS and the NLMS algorithms, they have been widely used in various applications [5].

Other studies on the variants of the LMS and NLMS algorithms have also been very active, among which their robust counterparts in impulsive noise environment are of great practical interests. This is because their performances will deteriorate significantly when the additive noise is of impulsive nature [6, 7], as both the LMS and NLMS algorithms assume that the additive noise is Gaussian distributed. Considerable efforts have been devoted to combat this adverse effect of impulsive noise on various adaptive filtering algorithms [7–12]. Among them, the least mean M-estimate (LMM) [7] algorithm, and its step size normalized version, the normalized least mean M-estimate

---

Part of this work was presented at *IEEE ISSPIT 2007* [46].

S. C. Chan · Y. Zhou (✉)  
Department of Electrical and Electronic Engineering,  
The University of Hong Kong,  
Pokfulam Road,  
Hong Kong, Hong Kong  
e-mail: yizhou@eee.hku.hk

S. C. Chan  
e-mail: scchan@eee.hku.hk

(NLMM) algorithm are two efficient generalizations of the LMS family. Like the recursive least M-estimate (RLM) algorithm [12], which is a robust variant of the RLS algorithm, they are derived from the concept of robust statistics techniques [13] where the M-estimate function [13] is minimized instead of the MSE to improve the robustness to impulsive noise. A common and important feature of these robust algorithms is the use of error nonlinearity to suppress the adverse effect of impulsive errors with large amplitude.

Because of the importance of these adaptive filtering algorithms, their convergence performance analyses have been long standing research problems. The convergence behavior of the LMS algorithm for Gaussian inputs was thoroughly studied in the classical work of Widrow et al. [1], in which the widely used concept of independence assumption was first introduced. Other related studies of the LMS algorithm with independent Gaussian inputs include [14–16]. On the other hand, the NLMS algorithm generally possesses an improved convergence speed over the LMS algorithm, but its analysis is more complicated due to the step size normalization. In [17] and [18], the mean and mean square behaviors of the NLMS algorithm for Gaussian inputs were studied. Analysis for independent Gaussian inputs in [19] also revealed some of the advantages of the NLMS algorithm over the LMS algorithm. Due to the difficulties in evaluating the expectations involved in the difference equations for the mean weight-error vector and its covariance matrix, and hence in deriving the general expressions for these equations, the works in [17] and [18] only concentrated on certain special cases of eigenvalue distribution of the input autocorrelation matrix. In [20] and [21], simplified input data models were introduced to facilitate the analysis so that useful analytical expressions can still be derived. In [22–24], the averaging principle was invoked to simplify the expectations involved in the difference equations. Basically, the normalization term is assumed to vary slowly with respect to the input correlation term and the power of the input vector is assumed to be either chi-square distributed with  $L$  degrees of freedom [22] or integrated otherwise [24]. Recently, Sayed et al. [25] proposed a unified framework based on the concept of energy conservation for analyzing the convergence of adaptive filtering algorithms. It has been applied to different adaptive filtering algorithms with satisfactory results [26, 27]. Other related works for the normalized sign-sign algorithm [28] and the NLMS algorithm in Gaussian noise environment with white input signal can be found in [29].

Convergence behavior of the RLM algorithm in contaminated Gaussian (CG) noise [30] using the modified Huber (MH) error nonlinearity was studied in [12]. However, a detailed analysis of the LMM and NLMM

algorithms both in Gaussian and CG additive noises is still unavailable, probably due to the complicated error nonlinearity and robust statistics involved, and the difficulty in analyzing the NLMS algorithm as mentioned above.

In this paper, we study the performance of a class of LMS and NLMS algorithms with error nonlinearity and Gaussian inputs and additive Gaussian as well as CG noise. In particular, the LMM and NLMM algorithms with MH error nonlinearity and adaptive threshold selection (ATS) will be studied in detail. It also includes a variety of other interesting algorithms and provides valuable insights into the advantages of using ATS and the effectiveness of the LMM/NLMM algorithms in impulsive noise environment. The analysis is divided into two main parts:

- (1) We extend the framework of [17] and [18] to analyze the NLMS algorithm with a class of error nonlinearity, which we called the M-nonlinearity, in Gaussian noise. This includes most M-estimate functions and several other commonly used error nonlinearities. Using the conventional Price's theorem [31–33], new decoupled difference equations describing the mean and mean square convergence behaviors are derived. The final results closely resemble the classical results for LMS in [1] and can be easily reduced to related works such as [34, 35]. Moreover, it is found that the nonlinearity will in general slow down the adaptation rate and the normalization process will always speed up the maximum convergence rate of the NLMS algorithms over their LMS counterparts if the eigenvalues of the input autocorrelation matrices are unequal. The convergence performances of the LMM and NLMM algorithms are studied in detail. The theoretical and practical importance of ATS is also explained and analyzed.
- (2) Instead of using the Price's theorem for Gaussian variates as an approximation in analyzing the RLM algorithm with MH nonlinearity in CG noise as in [12], we show that it is also applicable to Gaussian mixtures with very slight modification. In fact, the Price's theorem is applicable to each independent Gaussian component as pointed out, without proof, in a short note by Price [36]. Using this result and those in (1) above, the mean and mean square convergence behaviors of the LMS and NLMS algorithms with M-nonlinearity in CG noise are derived. The LMM and NLMM algorithms are then analyzed in detail. The analytical results suggest that the M-nonlinearity with ATS helps to suppress the impulsive noise in exchange for a slightly slower adaptation rate than that in the Gaussian noise case.

Another key to the above analysis is the introduction of certain special functions called the generalized Abelian

integral functions [37], which are generalizations of the Abelian integral functions [38]. They allow us to obtain decoupled analytical formulas by evaluating the expectations involved in the difference equations and help us interpret the convergence behaviors of the NLMS algorithms with M-nonlinearity. Particularly, these analytical results can be reduced to those for the conventional NLMS algorithms in Gaussian additive noise which agree with [17, 18], except that new expressions for the excess mean square error (EMSE), stability bound and difference equations in terms of the generalized Abelian integrals are obtained. For clarity of presentation and its practical importance, this particular case is separately treated in the companion paper [37]. All the above results can also be readily generalized to the simpler LMS case. Moreover, the new results also agree with the conventional LMS algorithm [1], the LMS algorithms with dual sign (DS) nonlinearity [34] and the error function (EF) nonlinearity [35]. Monte Carlo simulation results confirm that the NLMM algorithm offers improved robustness to impulsive noise over the NLMS algorithm, and are in good agreement with the theoretical analysis.

The rest of this paper is organized as follows: In Section 2, the NLMS algorithm is briefly reviewed and the NLMM algorithm is introduced. In Section 3, the proposed convergence performance analysis is presented. Simulation results are given in Section 4 and conclusions are drawn in Section 5.

## 2 Normalized Least Mean M-Estimate Algorithm

### 2.1 The NLMS Algorithm

Consider the adaptive system identification problem in Fig. 1 where an adaptive filter with coefficient or weight vector of order  $L$ ,  $\mathbf{W}(n) = [w_1(n), w_2(n), \dots, w_L(n)]^T$ , is used to model an unknown system with impulse response  $\mathbf{W}^* = [w_1, w_2, \dots, w_L]^T$ . Here,  $(\cdot)^T$  denotes the transpose of a vector or a matrix. The unknown system and the adaptive filter are simultaneously excited by the same input  $x(n)$ . The output of the unknown system  $d_0(n)$  is assumed to be

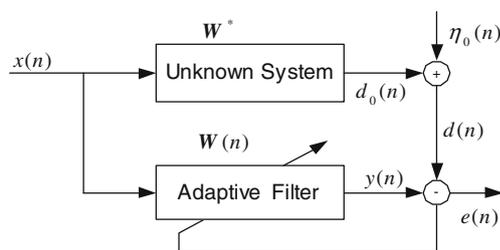


Figure 1 Adaptive system identification.

corrupted by a measurement noise  $\eta_0(n)$  to form the desired signal  $d(n)$  for the adaptive filter. The estimation error is given by  $e(n) = d(n) - y(n)$ . In the LMS algorithm, the MSE is minimized by updating the weight vector in the negative direction of the instantaneous gradient of the MSE with respect to the weight vector,  $-2e(n)\mathbf{X}(n)$ . This gives

$$\mathbf{W}(n + 1) = \mathbf{W}(n) + \mu e(n)\mathbf{X}(n), \tag{1}$$

where  $e(n) = d(n) - \mathbf{W}^T(n)\mathbf{X}(n)$ ,  $\mathbf{X}(n) = [x(n), x(n - 1), \dots, x(n - L + 1)]^T$  is the input vector at time instant  $n$  and  $\mu$  is a constant step size parameter chosen to reduce the gradient noise and to control the convergence rate and steady state error of the algorithm. In the NLMS algorithm, the step size is normalized by the energy of the input vector,  $\mathbf{X}^T(n)\mathbf{X}(n)$ , which gives:

$$\mathbf{W}(n + 1) = \mathbf{W}(n) + \frac{\mu e(n)\mathbf{X}(n)}{\varepsilon + \mathbf{X}^T(n)\mathbf{X}(n)}. \tag{2}$$

The step size  $\mu$  in the NLMS algorithm is a positive constant which should be chosen in the range  $0 < \mu < 2$  to ensure convergence of the algorithm.  $\varepsilon$  is a small positive value used to avoid division by zero.

When the additive noise  $\eta_0(n)$  is of impulsive nature, the performance of the NLMS algorithm which is based on the MSE criterion will deteriorate significantly. Recently, it is pointed out in [29] and [39] that the normalization mechanism of the NLMS algorithm can provide some degree of protection against impulsive noise. However, as will be illustrated in the subsequent computer experiments, the NLMS algorithm is still sensitive to consecutive impulses corrupting the desired signals. This motivates us to consider the NLMM algorithm.

### 2.2 The Normalized Least Mean M-Estimate Algorithm

Many techniques have been employed to reduce the hostile effect on the system due to impulsive interference. Examples include algorithms based on median filtering [8, 9], nonlinear clipping approaches [10, 11], and M-estimation-based algorithms such as the LMM [7] and RLM [12] algorithms. In particular, the last two algorithms were developed by minimizing the robust M-estimate cost functions instead of the conventional MSE criterion. Their improved robustness to impulsive noise and performance comparison with other relevant algorithms were thoroughly discussed in [7] and [12]. In the following, the concept of M-estimate cost function will be briefly reviewed and the NLMM algorithm will be derived.

In the LMM algorithm [7], the least mean M-estimate distortion measure  $J_\rho = E[\rho(e(n))]$  is minimized, where  $\rho(e)$  is an M-estimate function, which can be chosen as the Hampel’s three parts redescending function [40] or the

modified Huber (MH) M-estimate function [13] as shown in Fig. 2a:

$$\rho_{\text{MH}}(e) = \begin{cases} e^2/2, & 0 \leq |e| < \xi \\ \xi^2/2, & \text{otherwise,} \end{cases} \quad (3)$$

where  $\xi$  is the threshold parameter used to suppress the effect of outlier when the estimation error  $e$  is very large. Notice that when  $\rho(e) = e^2/2$  it reduces to the conventional MSE criterion. Like the LMS algorithm,  $J_p$  is minimized by updating  $\mathbf{W}(n)$  in the negative direction of the instantaneous gradient vector  $\widehat{\nabla}_{\mathbf{W}p}$ . Hence, the gradient vector,  $\nabla_{\mathbf{W}}(J_p)$ , is approximated by

$$\nabla_{\mathbf{W}}(J_p) = E \left[ -\frac{\partial \rho(e(n))}{\partial \mathbf{W}} \right] \approx \widehat{\nabla}_{\mathbf{W}p} = -\frac{\partial}{\partial \mathbf{W}} \rho(e(n)) = -q(e(n))e(n)\mathbf{X}(n), \quad (4)$$

where  $q(e) = \psi(e)/e = (\partial \rho(e)/\partial e)/e$  and  $\psi(e)$  is the score function. For the MH function:

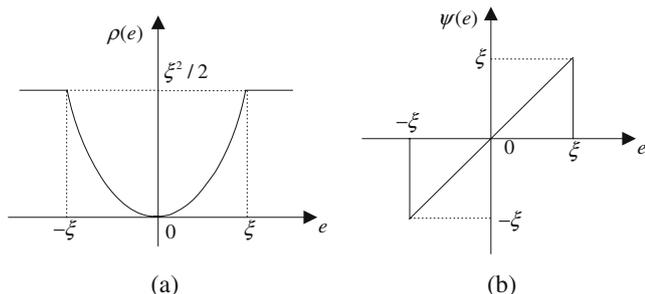
$$q_{\text{MH}}(e) = \frac{\psi_{\text{MH}}(e)}{e} = \begin{cases} 1, & 0 \leq |e| < \xi \\ 0, & \text{otherwise} \end{cases} \quad (5)$$

which is depicted in Fig. 2b.

Finally, we obtain the LMM algorithm as follows

$$\mathbf{W}(n+1) = \mathbf{W}(n) - \mu \widehat{\nabla}_{\mathbf{W}p} = \mathbf{W}(n) + \mu \psi(e(n))\mathbf{X}(n). \quad (6)$$

In general, when  $e(n)$  is smaller than  $\xi$ , the weight function  $q(e(n))$  is equal to one and (6) becomes identical to the LMS algorithm. When  $e(n)$  is larger than certain thresholds, say  $\xi$  in the MH function,  $q(e(n))$  will become zero and prevent the weight vector from updating. Thus the LMM algorithm can effectively reduce the adverse effect of large estimation error on the update of the filter coefficients. Another interpretation of (6) is that the error term is passed through a nonlinear device  $\psi(e)$ . This type of adaptive filtering algorithms has been studied previously for the error-function nonlinearity [35] and the sign nonlinearity [34]. The former concludes that the nonlinearity will slow down the convergence rate, while the latter is mainly introduced to reduce the implementation complexity. Both



**Figure 2** a The modified Huber M-estimate function  $\rho_{\text{MH}}(e)$ ; b  $\psi_{\text{MH}}(e)$ , the score function of  $\rho_{\text{MH}}(e)$ .

of them did not recognize the robustness of this class of algorithms to impulsive outliers. This was later studied by Koike in [10, 29] and using the clipping nonlinearity, and [6, 7, 41] using M-estimation. In [7], the threshold parameter  $\xi$  in the MH function is continuously updated using a technique called adaptive threshold selection (ATS), which greatly improves the convergence speed and steady state error.

*Adaptive Threshold Selection (ATS)* In ATS,  $e(n)$  is assumed to be Gaussian distributed except being corrupted occasionally by additive impulsive noise. By using the following robust variance estimate

$$\widehat{\sigma}_e^2(n) = \lambda_\sigma \widehat{\sigma}_e^2(n-1) + c_1(1 - \lambda_\sigma) \text{med}(A_e(n)), \quad (7)$$

$\widehat{\sigma}_e^2(n)$ , the variance of the “impulse-free” error  $\hat{e}(n)$ , can be estimated. Hence, it can be used to detect and reject the adverse outliers in  $e(n)$ . The forgetting factor  $\lambda_\sigma$  is a positive real number close to but smaller than one.  $\text{med}(\cdot)$  is the median operator.  $A_e(n) = \{e^2(n), \dots, e^2(n - N_w + 1)\}$ .  $c_1=2.13$  is the correction factor for median estimation and  $N_w$  is the length of the data set. The probability of  $|\hat{e}(n)|$  greater than a given threshold  $T_h$  is given by

$$\theta_T(n) = P_r\{|e(n)| > T_h\} = \text{erfc}\left(\frac{T_h}{\sqrt{2}\widehat{\sigma}_e(n)}\right), \quad (8)$$

where  $\text{erfc}(x) = (2/\sqrt{\pi}) \int_x^\infty e^{-t^2} dt$  is the complementary error function. Let  $\theta_\xi = P_r\{|e(n)| > \xi\}$  be the probabilities that  $e(n)$  is greater than  $\xi$ , the value of  $\xi$  can be determined by appropriate selection of  $\theta_\xi$  according to (8). For example, if  $\theta_\xi$  is chosen as 0.01, there will be 99% confidence to reject it when  $|\hat{e}(n)| > \xi$ . Accordingly,  $\xi$  can be obtained from (8) as

$$\xi = k_\xi \widehat{\sigma}_e(n) = 2.576 \widehat{\sigma}_e(n). \quad (9)$$

Similar expressions for the Hampel’s three parts re-descending function can be found in [7, 12]. This technique, which is referred to as adaptive threshold selection (ATS), is very important both theoretically and practically to the LMM and NLMM algorithms as we shall further elaborate in Sections 3 and 4. Some commonly used M-estimate functions, which we called the M-nonlinearity, and useful properties pertaining to their analyses in Section 3 are also summarized in Appendix C.

The computational complexity of the LMM algorithm per iteration is of order  $O(L)$  with additional  $O(N_w \log N_w)$  operations for performing the median filtering [7]. The window length  $N_w$  is usually chosen between 5 and 9.

If the step size  $\mu$  in (6) for the LMM algorithm is again normalized by the squared norm of the input vector as in

the NLMS algorithm, the following NLMM algorithm is obtained

$$\mathbf{W}(n+1) = \mathbf{W}(n) + \frac{\mu\psi(e(n))\mathbf{X}(n)}{\varepsilon + \mathbf{X}^T(n)\mathbf{X}(n)}, \tag{10}$$

where  $\psi(e(n)) = q(e(n))e(n)$ . Alternatively, (10) can be derived using the Least Perturbation Property (LPP) [5] originally proposed for the NLMS algorithm. Given the *a priori* output estimation error  $e(n) = d(n) - \mathbf{X}^T(n)\mathbf{W}(n)$ . The NLMS algorithm can be derived by seeking a  $\mathbf{W}(n+1)$  such that its deviation from  $\mathbf{W}(n)$  is minimized in the 2-norm, while satisfying a constraint between the a posteriori output estimation error  $e_p(n) = d(n) - \mathbf{X}^T(n)\mathbf{W}(n+1)$  and the prior output estimation error as follows:

$$\min_{\mathbf{W}(n+1)} \|\mathbf{W}(n+1) - \mathbf{W}(n)\|^2,$$

subject to

$$e_p(n) = \left(1 - \frac{\mu\|\mathbf{X}(n)\|^2}{\varepsilon + \|\mathbf{X}(n)\|^2}\right)e(n). \tag{11}$$

The solution of this problem yields the NLMS conventional update in (2). In M-estimation, the *a priori* output estimation error  $e(n)$  is replaced by the score function  $\psi(e(n))$  to suppress the adverse effect of impulsive noise. Substituting this into the constraint above and solving for  $\mathbf{W}(n+1)$  will yield the update in (10) above, since the two optimization problems have the same structure except for that  $e(n)$  is now replaced by  $\psi(e(n))$ .

As will be shown in the performance analysis in Section 3 and the simulation results in Section 4, this normalization brings faster convergence speed when the input is highly colored. Compared to the LMM algorithm for real inputs, the computational complexity is increased by one multiplication and two additions for updating  $\mathbf{X}^T(n)\mathbf{X}(n)$ , and one addition and one division for evaluating  $\mu/(\varepsilon + \mathbf{X}^T(n)\mathbf{X}(n))$ . The performance advantage of the NLMS algorithm for white inputs has been also analyzed in detail by Douglas et al. [42].

### 3 Mean and Mean Square Convergence Analysis

In this section, we first present a detailed convergence performance analysis of the NLMS algorithms with M-nonlinearity  $\psi(e)$  for Gaussian input and independent white Gaussian additive noise. More precisely, by extending the approach in [7, 12] and using the conventional Price’s theorem, it is possible to derive analytical expressions for modeling their mean and mean square convergence behaviors. Next, we extend the Price’s theorem to mixture Gaussian processes and, by using the above results, present

a detailed convergence performance analysis of these NLMS algorithms for Gaussian inputs and independent CG additive noise. In particular, the LMM and NLMM algorithms with MH function and ATS will be studied in detail. The final results are new expressions for EMSE, stability bound and difference equations describing the convergence behaviors of the various algorithms in terms of the generalized Abelian integral functions. They are similar to the classical results of the LMS algorithm by Widrow et al. [1], which allow us to clearly interpret and compare the convergence behaviors of this class of algorithms. To simplify the analysis, the following assumptions are made:

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

*Assumption 2:* The additive noise  $\eta_0(n)$  is assumed to be a Gaussian noise for the analysis in section A below ( $\eta_0(n)=\eta_g(n)$ ). For the analysis in Section B below,  $\eta_0(n)$  is modeled as a CG noise which is a frequently used model for analyzing impulsive noise. More precisely, it is given by:

$$\eta_0(n) = \eta_g(n) + \eta_{im}(n) = \eta_g(n) + b(n)\eta_w(n), \tag{12}$$

where  $\eta_g(n)$  and  $\eta_w(n)$  are both independent and identically distributed (i.i.d.) zero mean Gaussian sequences with respective variance  $\sigma_g^2$  and  $\sigma_w^2$ .  $b(n)$  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  $\eta_0(n)$  are then given by  $\sigma_{im}^2 = p_r\sigma_w^2$  and  $\sigma_{\eta_0}^2 = \sigma_g^2 + \sigma_{im}^2 = \sigma_g^2 + p_r\sigma_w^2$ . 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.  $\sigma_{\Sigma}^2 = \sigma_g^2 + \sigma_w^2$ . Accordingly, the probability distribution function (PDF) of this CG distribution is given by

$$f_{\eta_0}(\eta) = \frac{1-p_r}{\sqrt{2\pi\sigma_g^2}} \exp\left(-\frac{\eta^2}{2\sigma_g^2}\right) + \frac{p_r}{\sqrt{2\pi\sigma_{\Sigma}^2}} \exp\left(-\frac{\eta^2}{2\sigma_{\Sigma}^2}\right). \tag{13}$$

*Assumption 3:*  $\mathbf{W}(n)$ ,  $x(n)$  and  $\eta_0(n)$  are statistically independent (the independent assumption [1]). Although this assumption is not completely valid in general applications, it is a good approximation and is commonly used to simplify the convergence analysis of numerous adaptive filtering algorithms. Moreover, we denote  $\mathbf{W}^* = \mathbf{R}_{XX}^{-1}\mathbf{P}_{dX}$  as the optimal Wiener solution, where  $\mathbf{P}_{dX} = E[d(n)\mathbf{X}(n)]$  is the ensemble-averaged cross-correlation vector between  $\mathbf{X}(n)$  and  $d(n)$ .

### 3.1 Convergence Behaviors of the NLMS Algorithms with M-Nonlinearity and Gaussian Inputs and Noises

#### 3.1.1 Mean Behavior

From (10), the weight-error vector  $\mathbf{v}(n) = \mathbf{W}^* - \mathbf{W}(n)$  can be written as:

$$\mathbf{v}(n + 1) = \mathbf{v}(n) - \frac{\mu\psi(e(n))\mathbf{X}(n)}{\varepsilon + \mathbf{X}^T(n)\mathbf{X}(n)}. \tag{14}$$

If  $\psi(e(n))=e(n)$ , (14) will reduce to the conventional NLMS algorithm (2). Taking expectation on both sides of (14), we get

$$E[\mathbf{v}(n + 1)] = E[\mathbf{v}(n)] - \mu\mathbf{H}_1, \tag{15}$$

where  $\mathbf{H}_1 = E[\psi(e(n))\mathbf{X}(n)/(\varepsilon + \mathbf{X}^T(n)\mathbf{X}(n))]$  and  $E[\cdot]$  denotes the expectation over  $\{\mathbf{v}(n), \mathbf{X}(n), \eta_g(n)\}$  and is written more clearly as  $E_{\{\mathbf{v}, \mathbf{X}, \eta_g\}}[\cdot]$ . Since  $\mathbf{X}(n)$  and  $\eta_g(n)$  are stationary, we can drop the time index  $n$  in the expectation to get  $\mathbf{H}_1 = E[\psi(e)\mathbf{X}/(\varepsilon + \mathbf{X}^T\mathbf{X})]$ .

In the conventional NLMS algorithm [17] and [18], similar difference equation for the mean weight-error vector (c.f. [18, Eq. 11]) was obtained:

$$E[\mathbf{v}(n + 1)] = (\mathbf{I} - \mu\mathbf{F}_\varepsilon)E[\mathbf{v}(n)], \tag{16}$$

where  $\mathbf{F}_\varepsilon = E[\mathbf{X}(n)\mathbf{X}^T(n)/(\varepsilon + \mathbf{X}^T(n)\mathbf{X}(n))]$  and  $\mathbf{I}$  is the identity matrix. For convenience, the variables have been renamed according to the notation in this paper. Moreover,  $\mathbf{F}_\varepsilon$  was further diagonalized to  $\mathbf{H}_\varepsilon$  whose  $i$ -th element is  $[\mathbf{H}_\varepsilon]_{i,i} = \int_0^\infty \frac{\exp(-\beta\varepsilon)}{|\mathbf{I} + 2\beta\mathbf{R}_{XX}|^{1/2}} \frac{\lambda_i}{1 + 2\beta\lambda_i} d\beta$  (c. f. [18, Eq. 14]), where  $\lambda_i$  is the  $i$ -th eigenvalue of  $\mathbf{R}_{XX}$ . It was evaluated analytically in [17] for three important cases with different eigenvalue distributions (in [18], only the first case was elaborated): (1) white input signal with  $\lambda_1 = \dots = \lambda_L$ ; (2) two signal subspaces with equal powers  $\lambda_1 = \dots = \lambda_K = a$  and  $\lambda_{K+1} = \dots = \lambda_L = b$ ; (3) distinct pairs,  $\lambda_1 = \lambda_2, \lambda_3 = \lambda_4, \dots, \lambda_{L-1} = \lambda_L$  (assuming  $L$  is even). Besides these three special cases, no general solution to  $\mathbf{H}_\varepsilon$  was provided. Therefore, general closed-form formulas for modeling the mean and mean square behavior of NLMS algorithm were unavailable in [17] and [18]. To our best knowledge, no such analytical solution has ever appeared in the literature.

Here, we pursue another direction by treating some of these integrals involved as special functions and carry them throughout the analysis. Furthermore, using the Price’s theorem, the expectation involving the M-nonlinearity and ATS can be decoupled from the rest of the equations. The final formulas containing these special integral functions

still allow us to clearly interpret the convergence behavior of the NLMS algorithms with error nonlinearity. More precisely, it is shown in Appendix A (A-14) that

$$\mathbf{H}_1 \approx A_\psi(\sigma_e^2)\mathbf{U}\Lambda\mathbf{D}_\Lambda\mathbf{U}^TE[\mathbf{v}(n)]. \tag{17}$$

where  $A_\psi(\sigma_e^2) = \overline{\psi'}(\sigma_e^2) = \frac{1}{\sqrt{2\pi\sigma_e^2}} \int_{-\infty}^\infty \psi'(e) \exp(-e^2/(2\sigma_e^2)) de$ ,  $\mathbf{R}_{XX} = \mathbf{U}\Lambda\mathbf{U}^T$  is the eigenvalue decomposition of  $\mathbf{R}_{XX}$  and  $\Lambda = \text{diag}(\lambda_1, \lambda_2, \dots, \lambda_L)$  contains its eigenvalues.  $\mathbf{D}_\Lambda$  is a diagonal matrix with the  $i$ -th diagonal entry given by (A-16):  $[\mathbf{D}_\Lambda]_{i,i} = I_i(\Lambda) = \int_0^\infty \exp(-\beta\varepsilon) \left[ \prod_{k=1}^L (2\beta\lambda_k + 1)^{-1/2} \right] (2\beta\lambda_i + 1)^{-1} d\beta$ , which is a generalized Abelian integral function, whereas the conventional Abelian integral has the form  $I_a(x) = \int_0^x [q(\beta)]^{-1/2} d\beta$  with  $q(\beta)$  being a polynomial in  $\beta$ . It is also similar to  $[\mathbf{H}_\varepsilon]_{i,i}$  in [18].

Substituting (17) into (15), we obtain

$$E[\mathbf{v}(n + 1)] = (\mathbf{I} - \mu A_\psi(\sigma_e^2(n))\mathbf{U}\Lambda\mathbf{D}_\Lambda\mathbf{U}^T)E[\mathbf{v}(n)], \tag{18}$$

where  $\sigma_e^2(n) = E[\mathbf{v}^T(n)\mathbf{R}_{XX}\mathbf{v}(n)] + \sigma_g^2$  and the approximate sign has been replaced by the equality sign. (18) can also be written in the natural coordinate  $\mathbf{V}(n) = \mathbf{U}^T\mathbf{v}(n)$  as

$$E[\mathbf{V}(n + 1)] = (\mathbf{I} - \mu A_\psi(\sigma_e^2(n))\Lambda\mathbf{D}_\Lambda)E[\mathbf{V}(n)], \tag{19}$$

which is equivalent to  $L$  scalar first order finite difference equations as follows:

$$E[\mathbf{V}(n + 1)]_i = (1 - \mu A_\psi(\sigma_e^2(n))\lambda_i I_i(\Lambda))E[\mathbf{V}(n)]_i, \tag{20}$$

where  $E[\mathbf{V}(n)]_i$  is the  $i$ -th element of the vector  $E[\mathbf{V}(n)]$ .

Though the maximum possible step size is in general difficult to obtain, a sufficient condition for the algorithm to converge is  $|1 - \mu A_\psi(\sigma_e^2)\lambda_i I_i(\Lambda)| \leq 1$ , for all  $i$ . If  $\overline{\psi'}(\sigma_e^2)$  is bounded above by a constant  $\overline{\psi'}_{\max} = A_{\psi_{\max}}$ , then a conservative maximum step size bound is

$$\mu < \frac{2}{A_{\psi_{\max}}\lambda_{\max}I_{i,\lambda_{\max}}(\Lambda)} \tag{21}$$

which yields good estimates in practical algorithms such as the LMM, NLMM, and those in [34, 35].

#### Remarks

##### (R-A1): LMS algorithms with error nonlinearity:

As pointed out at the end of Appendix A, our analysis will reduce to the LMS case when  $\mathbf{D}_\Lambda = \mathbf{I}$  and Eq. (18) strictly holds for general error nonlinearity  $\psi(e)$ . Except for the conventional LS error criterion, (19) or (20) are generally a set of nonlinear difference equations due to the error nonlinearity. A general solution is rather difficult to obtain because the term  $A_\psi(\sigma_e^2) = \overline{\psi'}(\sigma_e^2)$  is dependent on the MSE. For the DS nonlinearity, (19) agrees with the

results in [34], and moreover, Mathews et al. showed that the resulting equation is convergent using the Doob’s theorem [43, 44]. (19) also agrees with the result in [35] for LMS algorithm with EF nonlinearity, in which the difference equation was approximated with a differential equation and it was showed that there are two possible adapting phases, the nonlinear and linear ones. If the nonlinearity  $\psi(e)$  is known, then it may be possible to derive its convergence behavior by converting it to a nonlinear ordinary differential equation (ODE).  $A_\psi(\sigma_e^2)$  for some commonly used error nonlinearities are summarized in Table 1.

For most M-estimate functions,  $\psi(e)=q(e)e$ .  $q(e)$  is equal to one when  $|e|$  is less than a certain threshold  $\xi$  and will gradually decrease to reduce its sensitivity to impulses with large amplitude. Hence,  $0 \leq \psi'(e) \leq 1$  and it is approximately equal to one when  $|e|$  is less than  $\xi$ . Specifically, for the LMM and NLMM algorithms using MH nonlinearity, it can be shown from (C-5) and (5) that:

$$A_{MH}(\sigma_e^2) = \text{erf}\left(\frac{\xi}{\sqrt{2}\sigma_e}\right) - \frac{2\xi}{\sqrt{2\pi}\sigma_e} \exp\left(-\frac{\xi^2}{2\sigma_e^2}\right),$$

with  $\lim_{\sigma_e^2 \rightarrow 0} A_{MH}(\sigma_e^2) \rightarrow 1$  and  $\lim_{\sigma_e^2 \rightarrow \infty} A_{MH}(\sigma_e^2) \rightarrow 0$ . For sufficiently small step size  $\mu$ , the algorithm will converge and  $\sigma_e^2$  will decrease. Because when  $\sigma_e^2$  is much larger than  $\xi^2$ ,  $A_{MH}(\sigma_e^2) \rightarrow 0$ ,  $\xi$  should be carefully chosen otherwise the step size can be unnecessarily small as adaptation starts. This is because  $\sigma_e^2$  can be much larger than  $\xi^2$  initially, which depends in turn on the initial condition. If  $A_{MH}(\sigma_e^2)$  is not made adaptive, an inappropriately chosen  $\xi$  may suppress the signal component, instead of just the outliers. This will cause  $A_{MH}(\sigma_e^2)$  to increase gradually and lead to the so called “nonlinear adaptation” problem encountered by the EF nonlinearity in [35]. When the algorithm is nearly converged,  $\sigma_e^2$  will be approximately constant and the asymptotic rate of convergence is  $1 - \mu A_{MH}(\sigma_e^2(\infty))I_i(\Lambda)$ , which is the so called “linear adaptation”.

Since nonlinear adaptation is typically slow, it should be avoided. In the LMM and NLMM algorithms,  $\xi$  is chosen to be a multiple of the estimated  $\sigma_e$  as shown in (9). If  $\hat{\sigma}_e^2 \approx \sigma_e^2$ , this helps to avoid significant signal suppression by maintaining a fairly stationary  $A_{MH}(\sigma_e^2)$  as follows:

$$A_{MH} \approx \text{erf}\left(\frac{k_\xi}{\sqrt{2}}\right) - \frac{2k_\xi}{\sqrt{2\pi}} \exp\left(-\frac{k_\xi^2}{2}\right),$$

which is approximately constant and is slightly less than one. From (20), we can see that the degradation in mean convergence over their LMS and NLMS counterparts is therefore minimal. When an impulse with large amplitude is encountered, we will momentarily have  $\sigma_e^2 \gg k_\xi^2 \hat{\sigma}_e^2$  and the measurement will be discarded to improve robustness.

In Appendix C, it is further shown that for most M-estimate functions and nonlinearities (c.f. (C-1) for the definition of M-nonlinearities),  $A_\psi(\sigma_e^2)$  is approximately

**Table 1** List of  $A_\psi(\sigma_e^2)$ ,  $B_\psi(\sigma_e^2)$  and  $C_\psi(\sigma_e^2)$  for three related algorithms.

Nonlinearity	$\psi(\varepsilon)$	$A_\psi(\sigma_e^2)$	$B_\psi(\sigma_e^2)$	$C_\psi(\sigma_e^2)$
Modified Huber	$\psi_{MH}(\varepsilon) = \begin{cases} \varepsilon, &  \varepsilon  \leq \xi \\ 0, & \text{otherwise} \end{cases}$	$\frac{2}{\sqrt{2\pi}\sigma_e} \left[ \int_0^\xi \exp\left(-\frac{u^2}{2\sigma_e^2}\right) du - \xi \exp\left(-\frac{\xi^2}{2\sigma_e^2}\right) \right]$	$\frac{2\sigma_e}{\sqrt{2\pi}} \left[ \int_0^\xi \exp\left(-\frac{u^2}{2\sigma_e^2}\right) du - \xi \exp\left(-\frac{\xi^2}{2\sigma_e^2}\right) \right]$	$\frac{2}{\sqrt{2\pi}\sigma_e} \left[ \int_0^\xi \exp\left(-\frac{u^2}{2\sigma_e^2}\right) du - \xi \exp\left(-\frac{\xi^2}{2\sigma_e^2}\right) \right] - \left(\frac{\xi}{\sqrt{2\pi}\sigma_e}\right) \exp\left(-\frac{\xi^2}{2\sigma_e^2}\right)$
Error function	$\psi_{EF}(\varepsilon) = \int_0^\varepsilon \exp\left(-u^2/2\sigma_y^2\right) du$	$\sigma_y / (\sigma_y^2 + \sigma_e^2)^{1/2}$	$\sigma_y^2 \sin^{-1}\left(\frac{1}{1+(\sigma_y^2/\sigma_e^2)}\right)$	$\left[ \left(1 + 2\sigma_e^2/\sigma_y^2\right) \left(1 + \sigma_e^2/\sigma_y^2\right) \right]^{-1} - \frac{1}{2} \exp\left(-(\tau^2/2\sigma_e^2)\right) (L^2 - 1) (\tau/\sigma_e)$
Dual sign	$\psi_{DS}(\varepsilon) = \begin{cases} \text{sgn}(\varepsilon), &  \varepsilon  \leq \tau \\ L \text{sgn}(\varepsilon), &  \varepsilon  > \tau \end{cases}$	$\sqrt{\frac{2}{\pi}} \frac{1}{\sigma_e} [1 + (L-1) \exp(-\tau^2/2\sigma_e^2)]$	$L^2 - (L^2 - 1) \sqrt{\frac{2}{\pi}} \int_0^\tau \exp(-u^2/2) du$	

independent of  $\sigma_e^2$  if ATS mentioned above is used. Consequently, the convergence rate is approximately constant and similar to that of the LMS and NLMS algorithms. Because of its good performance, ATS is highly recommended in practical applications.

**(R-A2): NLMS algorithms with error nonlinearity:**

In the normalized case, (19) provides a good approximation at the steady state of the algorithm and for the NLMS algorithms using the M-nonlinearities together with ATS, which we have elaborated above (c.f. the approximation used in Appendix A and the justification at Appendix C). It should be noted that no such approximation is used in the variants of the LMS algorithms [34, 35] mentioned above. If ATS is not employed, (19) is only an approximation and nonlinear adaptation is likely to be encountered, depending on the clipping level of the nonlinearity and variances of the error signals. The adaptation is typically slow and accurate analytical solution is very difficult to obtain. When the algorithm is nearly converged,  $\sigma_e^2$  will be approximately constant and (19) will be a good approximation. The asymptotic rate of convergence is  $1 - \mu A_{MH}(\sigma_e^2(\infty))I_i(\Lambda)$ , which is the familiar “linear adaptation” phase. To avoid slow nonlinear adaptation, ATS is again recommended in practical applications. A conservative maximum step size can also be estimated from (21). Because of its importance, the main focus of this paper is those algorithms employing ATS.

**(R-A3): Comparison of LMS and NLMS algorithms with M-nonlinearity and ATS:**

Next, we briefly compare the LMS/NLMS-based algorithms. It can be seen that with M-nonlinearity and ATS, the convergence rate of the NLMS-based algorithms is  $1 - \mu A_\psi(\sigma_e^2(n))I_i(\Lambda)$ , where  $A_\psi(\sigma_e^2(n))$  is approximately constant and usually has a value slightly smaller than one when no impulses are encountered. As a result, compared with the conventional LMS-based algorithms, the step size of the normalized algorithms is changed by a factor of  $1/(A_\psi(\sigma_e^2(n))I_{i,\lambda_{\max}}(\Lambda)) \approx 1/I_{i,\lambda_{\max}}(\Lambda)$ . Since the maximum of the product

$$\begin{aligned} \lambda_i I_i(\Lambda) &= \lambda_i \int_0^\infty \exp(-\beta\varepsilon) \left[ \prod_{k=1}^L (2\beta\lambda_k + 1)^{-1/2} \right] (2\beta\lambda_i + 1)^{-1} d\beta \\ &= \int_0^\infty \exp(-\beta\varepsilon) \left[ \prod_{k=1}^L (2\beta\lambda_k + 1)^{-1/2} \right] \left[ 2\beta + (\lambda_i)^{-1} \right]^{-1} d\beta, \end{aligned}$$

also achieves its maximum at the largest eigenvalue  $\lambda_{\max}$  with the corresponding value of  $I_i(\Lambda)$  given by  $I_{i,\lambda_{\max}}(\Lambda)$ , the fastest convergence rate of the normalized algorithm occurs when  $\mu = 1/(A_\psi(\sigma_e^2)\lambda_{\max}I_{i,\lambda_{\max}}(\Lambda))$  and it is limited by the mode corresponding to the smallest eigenvalue

$\lambda_{\min}$  with the corresponding value of  $I_i(\Lambda)$  given by  $I_{i,\lambda_{\min}}(\Lambda)$ . That is

$$1 - \frac{2\lambda_{\min}I_{i,\lambda_{\min}}(\Lambda)}{\lambda_{\max}I_{i,\lambda_{\max}}(\Lambda)}. \tag{22}$$

From the definition of  $I_i(\Lambda)$ , it can be shown that  $I_{i,\lambda_{\min}}(\Lambda)/I_{i,\lambda_{\max}}(\Lambda) \geq 1$ . 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 maximum convergence rate of the normalized algorithms using ATS is faster than the LMS-based algorithms if the eigenvalues are unequal. Similar conclusion is obtained for the conventional NLMS algorithm [37, 46].

*3.1.2 Mean Square Behavior*

Post-multiplying (14) by its transpose and taking expectation gives

$$\Xi(n+1) = \Xi(n) - \mathbf{M}_1 - \mathbf{M}_2 + \mathbf{M}_3, \tag{23}$$

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

$$\mathbf{M}_1 = \mu E_{\{\mathbf{v}\}}[\mathbf{H}\mathbf{v}^T] \approx \mu A_\psi(\sigma_e^2)\mathbf{U}\Lambda\mathbf{D}_\Lambda\mathbf{U}^T\Xi(n-1), \tag{24}$$

$$\mathbf{M}_2 = \mathbf{M}_1^T \approx \mu A_\psi(\sigma_e^2)\Xi(n)\mathbf{U}\mathbf{D}_\Lambda\Lambda\mathbf{U}^T, \tag{25}$$

and

$$\mathbf{M}_3 = \mu^2 E[\psi^2(e)\mathbf{X}\mathbf{X}^T/(\varepsilon + \mathbf{X}^T\mathbf{X})^2]. \tag{26}$$

Note, the expressions in (24) and (25) are obtained by using previous result in (18), and (26) is obtained from the stationarity and independence assumptions.  $\mathbf{M}_3$  is evaluated in Appendix B to be

$$\begin{aligned} \mathbf{M}_3 \approx & 2\mu^2 C_\psi(\sigma_e^2)\mathbf{U}\{\Lambda[(\mathbf{U}^T\Xi(n)\mathbf{U}) \circ \mathbf{I}(\Lambda)]\Lambda\}\mathbf{U}^T \\ & + \mu^2 S_\psi(\sigma_e^2)\mathbf{U}\bar{\mathbf{D}}_2\mathbf{U}^T + \mu^2 S_\psi(\sigma_e^2)\sigma_e^2\mathbf{U}\Lambda\mathbf{I}'(\Lambda)\mathbf{U}^T. \end{aligned} \tag{27}$$

where  $C_\psi(\sigma_e^2) = \frac{d}{d\sigma_e^2}E[\psi^2(e)]$ ,  $S_\psi(\sigma_e^2) = B_\psi(\sigma_e^2)/\sigma_e^2$ ,  $B_\psi(\sigma_e^2) = E[\psi^2(e)] = \frac{1}{\sqrt{2\pi}\sigma_e} \int_{-\infty}^\infty \psi^2(e) \exp\left(-\frac{e^2}{2\sigma_e^2}\right) de$ .  $\circ$  denotes element-wise product of two matrices (Hadamard product), and  $\mathbf{I}(\Lambda)$  and  $\mathbf{I}'(\Lambda)$  are defined in (B-14) and (B-16). The diagonal matrix  $\bar{\mathbf{D}}_2$  results from (B-14b) and its  $i$ -th element is  $[\bar{\mathbf{D}}_2]_{i,i} = \sum \lambda_k \lambda_i I_{ki}(\Lambda) [\mathbf{U}^T\Xi(n)\mathbf{U}]_{k,k}$ . For a given distortion measure  $\rho^k(e)$  and hence  $\psi(e)$ , these integrals can either be computed analytically or numerically.

Substituting (24)–(27) into (23), and using the natural coordinate  $\Phi(n) = \mathbf{U}^T\Xi(n)\mathbf{U}$ , one gets

$$\begin{aligned} \Phi(n+1) \approx & \Phi(n) - \mu A_\psi(\sigma_e^2) \Lambda \mathbf{D}_\Lambda \Phi(n) - \mu A_\psi(\sigma_e^2) \Phi(n) \mathbf{D}_\Lambda \Lambda \\ & + 2\mu^2 C_\psi(\sigma_e^2) [\Lambda(\Phi(n) \circ I(\Lambda))\Lambda] + \mu^2 S_\psi(\sigma_e^2) \tilde{\mathbf{D}}_2 \\ & + \mu^2 S_\psi(\sigma_e^2) \sigma_g^2 \Lambda \mathbf{I}'(\Lambda), \end{aligned} \tag{28}$$

where  $[\tilde{\mathbf{D}}_2]_{i,i} = \sum_k \lambda_k \lambda_i I_{ki}(\Lambda) [\Phi(n)]_{k,k}$ . (28) can be written as the following scalar form:

$$\begin{aligned} \Phi_{i,i}(n+1) \approx & (1 - 2\mu A_\psi(\sigma_e^2) \lambda_i I_i(\Lambda) + 2\mu^2 C_\psi(\sigma_e^2) \lambda_i^2 I_{ii}(\Lambda)) \Phi_{i,i}(n) \\ & + \mu^2 S_\psi(\sigma_e^2) \sum_k \lambda_k \lambda_i I_{ki}(\Lambda) \Phi_{k,k}(n) + \mu^2 S_\psi(\sigma_e^2) \sigma_g^2 \lambda_i I'_i(\Lambda). \end{aligned} \tag{29}$$

To study the step size bound for mean square convergence, we first note that the EMSE at time instant  $n$  is given by  $\text{EMSE}(n) = \text{Tr}(\Phi(n)\Lambda)$ . Assuming that algorithm converges, it is shown in Appendix B that the last two terms on the right hand side of (29) are tightly upper bounded by  $\mu^2 S_\psi(\sigma_e^2) \sigma_e^2(\infty) \lambda_i I'_i(\Lambda)$  at the steady state. Therefore, from (28), one gets an upper bound for the EMSE as

$$\begin{aligned} \text{EMSE}_{\text{NLMS-}\psi}(\infty) &= \text{Tr}(\Phi(\infty)\Lambda) \tag{30} \\ &\approx \frac{1}{2} \mu \sigma_e^2(\infty) \phi_{\text{NLMS-}\psi}, \end{aligned}$$

where  $\phi_{\text{NLMS-}\psi} = S_\psi(\sigma_e^2(\infty)) \sum_{i=1}^L \frac{\lambda_i I'_i(\Lambda)}{A_\psi(\sigma_e^2(\infty)) I_i(\Lambda) - \mu C_\psi(\sigma_e^2(\infty)) \lambda_i I_{ii}(\Lambda)}$ .

Using the fact that  $\sigma_e^2(\infty) = \text{EMSE}_{\text{NLMS-}\psi}(\infty) + \sigma_g^2$ , one gets

$$\text{EMSE}_{\text{NLMS-}\psi}(\infty) = \frac{\frac{1}{2} \mu \sigma_g^2 \phi_{\text{NLMS-}\psi}}{1 - \frac{1}{2} \mu \phi_{\text{NLMS-}\psi}}. \tag{31}$$

which is a nonlinear equation in  $\sigma_e^2(\infty)$  and hence  $\text{EMSE}_{\text{NLMS-}\psi}(\infty)$ . It can be seen that  $\text{EMSE}_{\text{NLMS-}\psi}(\infty)$  is unbounded when either its denominator becomes zero or when  $\phi_{\text{NLMS-}\psi}$  becomes unbounded due to any of the denominators of its partial sum becomes zero. These two conditions allow us to determine the following conditions for the maximum step size:

$$\mu < 2/\phi_{\text{NLMS-}\psi}, \tag{32a}$$

$$0 < \mu < A_\psi(\sigma_e^2(\infty)) I_i(\Lambda) / [C_\psi(\sigma_e^2(\infty)) \lambda_i I_{ii}(\Lambda)]. \tag{32b}$$

For the conventional LMS algorithm,  $A_\psi, C_\psi, I_i(\Lambda), I'_i(\Lambda)$ , and  $I_{ii}(\Lambda)$  are all equal to one, (32a) and (32b) are identical to the necessary and sufficient conditions for the mean square convergence of the LMS algorithm previously obtained in [14]. Similar results are obtained in [15] by solving the difference equation in  $\Phi(n)$  and in [16] by a matrix analysis technique. Furthermore, a lower bound of the maximum step size for mean square convergence is obtained in [15]. Here, we extend this approach to study the

stability bound of our algorithms. To this end, we write (32a) in full as follows

$$S_\psi(\sigma_e^2(\infty)) \sum_{i=1}^L \frac{\mu \lambda_i I'_i(\Lambda)}{A_\psi(\sigma_e^2(\infty)) I_i(\Lambda) - \mu C_\psi(\sigma_e^2(\infty)) \lambda_i I_{ii}(\Lambda)} < 2. \tag{33}$$

For convenience, rewrite (33) as

$$\sum_{i=1}^L \frac{\mu \lambda_i c_i}{1 - \mu \lambda_i d_i} < 2, \tag{34}$$

where  $c_i = I'_i(\Lambda) / (A_\psi(\sigma_e^2(\infty)) I_i(\Lambda))$ , and  $d_i = C_\psi(\sigma_e^2(\infty)) I_{ii}(\Lambda) / (A_\psi(\sigma_e^2(\infty)) I_i(\Lambda))$ .

The right hand side has singularities at  $1/(\lambda_i d_i)$ . Between 0 and  $\min(1/(\lambda_i d_i))$ , it is a monotonic increasing function of  $\mu$ . Therefore, it will reach the critical value of two on the right hand side of the equality at the smallest root  $\mu_{\max}$  of the equation

$$\sum_{i=1}^L \frac{\mu \lambda_i c_i}{1 - \mu \lambda_i d_i} = 2, \tag{35}$$

before  $\min(1/(\lambda_i d_i))$ . Any positive step size  $\mu$  below  $\mu_{\max}$  will ensure the mean square convergence of the LMS algorithm.

Let  $u = 2\mu^{-1}$  and rewrite (35) as

$$\ell(u) - \sum_{i=1}^L \lambda_i c_i \ell_i(u) = \prod_{i=1}^L (u - \bar{u}_i) = \sum_{i=1}^L (-1)^{L-i} b_{L-i} u^i = 0, \tag{36}$$

where,  $\ell(u) = \prod_{i=1}^L (u - 2\lambda_i d_i)$ ,  $\ell_i(u) = \ell(u) / (u - 2\lambda_i d_i)$ , and  $\bar{u}_i^{-1}$  are the roots of (36). The largest root of (36) (smallest root of (34)) is upper bounded (lower bounded) by [47]

$$u_{N-\max} \leq \frac{1}{L} \left[ s_1 + \sqrt{(L-1)(Ls_2 - s_1^2)} \right], \tag{37}$$

where  $s_1 = \sum_{i=1}^L \bar{u}_i = b_1$  and  $s_2 = \sum_{i=1}^L \bar{u}_i^2 = b_1^2 - 2b_2$ .

By comparing the coefficients on different sides of (36), one also gets

$$b_1 = \sum_{i=1}^L \lambda_i (2d_i + c_i)$$

and

$$b_2 = 4 \sum_{1 \leq i \neq j \leq L} \lambda_i \lambda_j d_i d_j + \sum_{i=1}^L \lambda_i c_i \left( 2 \sum_{1 \leq j \neq i \leq L} \lambda_j d_j \right). \tag{38}$$

From (37), a more convenient lower bound of  $\mu_{\max}$  can be obtained as follows

$$\begin{aligned} \mu_{\max} &\geq \frac{2L}{b_1 + \sqrt{(L-1)^2 b_1^2 - 2L(L-1)b_2}} \\ &\geq \frac{2L}{b_1 + \sqrt{(L-1)^2 b_1^2}} = 2/b_1 \equiv \mu_B \\ &= \frac{2A_\psi(\sigma_e^2(\infty))}{S_\psi(\sigma_e^2(\infty)) \sum_{i=1}^L \lambda_i [I'_i(\Lambda) / (I_i(\Lambda)) + 2C_\psi(\sigma_e^2(\infty)) I_{ii}(\Lambda) / I_i(\Lambda)]}. \end{aligned} \tag{39}$$

It can be seen from (39) that  $A_\psi(\sigma_e^2(\infty))$ ,  $C_\psi(\sigma_e^2(\infty))$  and  $S_\psi(\sigma_e^2(\infty))$  also depend on  $\sigma_e^2(\infty)$  and the exact step size bound is still very difficult to obtain analytically. For M-nonlinearities with ATS, they are approximately constant and the step size bound can be determined accordingly from  $\mu_B$ . Alternatively, one can replace  $A_\psi(\sigma_e^2(\infty))$ ,  $C_\psi(\sigma_e^2(\infty))$  and  $S_\psi(\sigma_e^2(\infty))$  by their worse case values to estimate the step size bound  $\mu_B$ .

In passing, we find from simulation results that the term  $\mu_k^2 \Sigma \lambda_k \lambda_i I_{ki}(\Lambda) \Phi_{k,k}(n)$  in (29) is very small for small EMSE. Consequently, (29) can be approximated as:

$$\Phi_{i,i}(n+1) \approx (1 - 2\mu A_\psi(\sigma_e^2) \lambda_i I_i(\Lambda) + 2\mu^2 C_\psi(\sigma_e^2) \lambda_i^2 I_{ii}(\Lambda)) \Phi_{i,i}(n) + \mu^2 S_\psi(\sigma_e^2) \sigma_g^2 \lambda_i I'_i(\Lambda).$$

For notation convenience, we have dropped the time index  $n$  in  $\sigma_e^2(n) = E[\mathbf{v}^T(n) \mathbf{R}_{XX} \mathbf{v}(n)] + \sigma_g^2$ . The algorithm will converge when  $|1 - 2\mu A_\psi(\sigma_e^2) \lambda_i I_i(\Lambda) + 2\mu^2 C_\psi(\sigma_e^2) \lambda_i^2 I_{ii}(\Lambda)| < 1$ , which gives (32b):  $\mu < A_\psi(\sigma_e^2) I_i(\Lambda) / (C_\psi(\sigma_e^2) \lambda_i I_{ii}(\Lambda))$  for all  $i$ . That is, the results of (32a) and (32b) are very close to each other. From the definition of  $I_i(\Lambda)$  and  $I_{ii}(\Lambda)$ , we then have  $I_i(\Lambda) / (\lambda_i I_{ii}(\Lambda)) = 2 / (1 - I'_i(\Lambda) / I_i(\Lambda)) > 2$ , where  $I'_i(\Lambda) = \int_0^\infty \exp(-\beta \varepsilon) \left[ \prod_{k=1}^L (2\beta \lambda_k + 1)^{-1/2} \right] (2\beta \lambda_i + 1)^{-2} d\beta$ . Therefore, a conservative stability bound for small EMSE is  $\mu_B < 2A_\psi(\sigma_e^2) / C_\psi(\sigma_e^2)$ . For M-nonlinearities with ATS,  $\mu_B < 2A_\psi / C_\psi$ , which is a useful “rule of thumb” step size bound because it does not require any prior knowledge of the input signal.

*Remarks*

**(R-A4): LMS algorithms with error nonlinearity:**

When  $I_i(\Lambda) = I'_i(\Lambda) = I_{ii}(\Lambda) = 1$ , our analysis will reduce to that for the LMS algorithm with general nonlinearity. (31) will reduce to

$$\text{EMSE}_{\text{LMS-}\psi}(\infty) \approx \frac{\frac{1}{2} \mu \sigma_g^2 \phi_{\text{LMS-}\psi}}{1 - \frac{1}{2} \mu \phi_{\text{LMS-}\psi}}, \tag{40}$$

where  $\phi_{\text{LMS-}\psi} = S_\psi(\sigma_e^2(\infty)) \sum_{i=1}^L \frac{\lambda_i}{A_\psi(\sigma_e^2(\infty)) - \mu C_\psi(\sigma_e^2(\infty)) \lambda_i}$ ,  $B_\psi(\sigma_e^2) = \sigma_e^2$  and  $S_\psi(\sigma_e^2)$  and  $C_\psi(\sigma_e^2)$  for some related algorithms are summarized in Table 1. For the DS algorithm [34], (40) will reduce to [34, Eq. (24)]. If  $\mu$  is sufficiently small, then the contribution of the weight-error vector to  $\sigma_e^2(\infty)$  can be ignored. Accordingly,  $\sigma_e^2(\infty) \approx \sigma_g^2$  and (31) can be simplified to  $\text{EMSE}_{\text{LMS-}\psi}(\infty) \approx \frac{\mu S_\psi(\sigma_e^2(\infty)) \sigma_g^2}{2A_\psi(\sigma_e^2(\infty))} \sum_{i=1}^L \lambda_i$ . This agrees with [34, Eq. (25)] for the DS algorithm. Further, if the input is white, it will reduce to [35 Eq. 45] for the EF nonlinearity case.

For M-nonlinearities with ATS,  $A_\psi(\sigma_e^2)$ ,  $S_\psi(\sigma_e^2)$  and  $C_\psi(\sigma_e^2)$  are approximately constant (c.f. Appendix C).

Denote them by  $A_\psi$ ,  $S_\psi$  and  $C_\psi$ , respectively. Then, (30) becomes

$$\text{EMSE}_{\text{LMS-}\psi}(\infty) \approx \frac{\frac{1}{2} \mu \sigma_g^2 \phi_{\text{LMS-}\psi}}{1 - \frac{1}{2} \mu \phi_{\text{LMS-}\psi}}, \tag{41}$$

where  $\phi_{\text{LMS-}\psi} = S_\psi \sum_{i=1}^L \frac{\lambda_i}{A_\psi - \mu \lambda_i C_\psi}$ . The stability bound from (32) is  $\mu < \min(2 / \phi_{\text{LMS-}\psi}, A_\psi / (C_\psi \lambda_i))$ , from which we can obtain its lower bound from (38) as follows:

$$\mu_{B\text{-LMS-}\psi} = \frac{2A_\psi}{S_\psi \sum_{i=1}^L \lambda_i (1 + 2C_\psi)}. \tag{42}$$

If  $\psi(e)=e$  and  $A_\psi = S_\psi = C_\psi = 1$  we obtain the traditional LMS algorithm. (40) and (42) will respectively reduce to the  $\text{EMSE}(\infty)$  and stability bound for the conventional LMS algorithm derived in [15]. For the LMM algorithm using MH-nonlinearity with a practical value of  $k_\xi = 2.576$ ,  $A_\psi$ ,  $S_\psi$ , and  $C_\psi$  are quite close to one, and its performance is therefore similar to that of the conventional LMS algorithm.

**(R-A5): NLMS algorithms with error nonlinearity:**

For the normalized version, (31) will be a good approximation at the steady state of the algorithm and for the NLMS algorithms using M-nonlinearities and ATS (c.f. the approximation used in Appendix B). For the latter, (31) can be simplified to

$$\text{EMSE}_{\text{NLMS-}\psi}(\infty) \approx \frac{\mu \sigma_e^2(\infty)}{2} \phi_{\text{NLMS-}\psi}, \tag{43}$$

where  $\phi_{\text{NLMS-}\psi} = \frac{S_\psi}{A_\psi} \sum_{i=1}^L \frac{\lambda_i I'_i(\Lambda)}{I_i(\Lambda) - \mu \lambda_i I_{ii}(\Lambda) (C_\psi / A_\psi)}$ . Solving (43) yields

$$\text{EMSE}_{\text{NLMS-}\psi}(\infty) \approx \frac{\frac{1}{2} \mu \sigma_g^2 \phi_{\text{NLMS-}\psi}}{1 - \frac{1}{2} \mu \phi_{\text{NLMS-}\psi}}. \tag{44}$$

The stability bound from (32) satisfies:  $\mu < 2 / \phi_{\text{NLMS-}\psi}$ ,  $0 < \mu < A_\psi / (C_\psi \lambda_i)$ , from which we can obtain its lower bound from (39) as follows:

$$\mu_{B\text{-NLMS-}\psi} = \frac{2A_\psi}{S_\psi \sum_{i=1}^L \lambda_i [I'_i(\Lambda) / I_i(\Lambda) + 2C_\psi I_{ii}(\Lambda) / I_i(\Lambda)]}. \tag{45}$$

For  $\psi(e)=e$  and  $A_\psi = S_\psi = C_\psi = 1$  we obtain the traditional NLMS algorithm. (43) and (45) will reduce to the  $\text{EMSE}(\infty)$  and stability bound for the conventional NLMS algorithm derived in [37]. For the NLMM algorithm using MH-nonlinearity with a practical value of  $k_\xi = 2.576$ ,  $A_\psi$ ,  $S_\psi$ , and  $C_\psi$  are quite close to one, and its performance is therefore similar to that of the conventional NLMS algorithm.

### 3.2 Convergence Behaviors of the NLMS Algorithm with Error Nonlinearity and Gaussian Input and CG Noise

We now analyze the mean and mean square behaviors of the various algorithms studied above in Section 3.1 in CG noise environment. From the previous analysis, we note that the assumption of Gaussian input and additive noise allows us to use the Price’s theorem to approximately decouple the effect of the nonlinearity. For most M-estimate functions which suppress outliers with large amplitude, the convergence rate will be slightly impaired with a similar EMSE, after using ATS. We shall show in the following that this will be paid off by their improved robustness to impulsive noise. Apparently, the Price’s theorem does not apply to Gaussian mixture. However, as pointed out in [36] and to be explained below, it is actually applicable to individual components of the mixture. Moreover, for the LMM and NLMM algorithms, the error signal is nearly Gaussian distributed after the impulsive component is suppressed. This also explains why the approximations in [12, 41] give excellent agreement with Monte Carlo simulations. A detailed analysis will be given below.

#### 3.2.1 Mean Behavior

Assume the additive noise  $\eta_0$  is now a CG noise as defined in (12), it is a Gaussian mixture consisting of two components  $\eta_{0\_1}$  and  $\eta_{0\_2}$ , each with zero mean and variance  $\sigma_1^2 = \sigma_g^2$  and  $\sigma_2^2 = \sigma_\Sigma^2$ , respectively. The occurrence probability of the impulsive noise is  $p_r$ . Accordingly,

$$E_{\{v, X, \eta_0\}}[f(\mathbf{X}(n), e(n))] = (1 - p_r)E_{\{v, X, \eta_{0\_1}\}}[f(\mathbf{X}(n), e(n))] + p_r E_{\{v, X, \eta_{0\_2}\}}[f(\mathbf{X}(n), e(n))], \tag{46}$$

where  $f(\mathbf{X}(n), e(n))$  is an arbitrary quantity whose statistical average is to be evaluated. Since  $\mathbf{X}(n)$ ,  $\eta_{0\_1}$ , and  $\eta_{0\_2}$  are Gaussian distributed, each of the expectation on the right hand side can be evaluated using the Price’s theorem. Consequently, the results in Section 3.1 can be carried forward to the CG case by firstly changing the noise power respectively to  $\sigma_g^2$  and  $\sigma_\Sigma^2$ , and then combining the two results using (46).

Recall the relation of the mean weight-error vector in (15):

$$E[\mathbf{v}(n+1)] = E[\mathbf{v}(n)] - \mu \mathbf{H}' \tag{47}$$

where  $\mathbf{H}' = E_{\{v, X, \eta_0\}} \left[ \frac{\psi(e(n))\mathbf{X}(n)}{\varepsilon + \mathbf{X}^T(n)\mathbf{X}(n)} \right] = (1 - p_r)\mathbf{H}'_1 + p_r\mathbf{H}'_2$ ,  $\mathbf{H}'_1$  and  $\mathbf{H}'_2$  are the expectation of the term inside the brackets above with respect to  $\{v, \mathbf{X}, \eta_{0\_1}\}$ , and  $\{v, \mathbf{X}, \eta_{0\_2}\}$ . From (17), we have  $\mathbf{H}'_i \approx A_\psi(\sigma_{e_i}^2) \mathbf{U} \Lambda \mathbf{D}_\Lambda \mathbf{U}^T E[\mathbf{v}(n)]$ ,  $i=1, 2$ , where  $\sigma_{e_1}^2 = \sigma_g^2$  and  $\sigma_{e_2}^2 = \sigma_\Sigma^2$ . Hence

$$\mathbf{H}' \approx \tilde{A}_\psi(\sigma_{e_g}^2, \sigma_{e_\Sigma}^2) \mathbf{U} \Lambda \mathbf{D}_\Lambda \mathbf{U}^T E[\mathbf{v}(n)], \tag{48}$$

where  $\sigma_{e_g}^2 = E[\mathbf{v}^T(n)\mathbf{R}_{XX}\mathbf{v}(n)] + \sigma_g^2$ ,  $\sigma_{e_\Sigma}^2 = E[\mathbf{v}^T(n)\mathbf{R}_{XX}\mathbf{v}(n)] + \sigma_\Sigma^2$ ,  $\tilde{A}_\psi(\sigma_{e_g}^2, \sigma_{e_\Sigma}^2) = (1 - p_r)A_\psi(\sigma_{e_g}^2) + p_r A_\psi(\sigma_{e_\Sigma}^2)$ , and  $\mathbf{U}$ ,  $\Lambda$ , and  $\mathbf{D}_\Lambda$  have been defined in Section 3.1. Substituting (18) into (46) and using the natural coordinate  $\mathbf{V}(n) = \mathbf{U}^T \mathbf{v}(n)$ , one gets

$$E[\mathbf{V}(n+1)]_i = \left(1 - \mu \tilde{A}_\psi(\sigma_{e_g}^2, \sigma_{e_\Sigma}^2) \lambda_i I_i(\Lambda)\right) E[\mathbf{V}(n)]_i. \tag{49}$$

For notational convenience, we have replaced the approximate symbol by the equality symbol. This yields the same form as (20), except for  $\tilde{A}_\psi(\sigma_{e_g}^2, \sigma_{e_\Sigma}^2)$ . Similar argument regarding the mean convergence in Section 3.1.1 also applies to (49). A sufficient condition for the algorithm to converge is  $|1 - \mu \tilde{A}_\psi(\sigma_{e_g}^2, \sigma_{e_\Sigma}^2) \lambda_i I_i(\Lambda)| < 1$ , for all  $i$ . If  $A_\psi(\sigma_e^2)$  is bounded above by  $A_{\psi\_max}$ , then following the argument in Section 3.1.1, the following conservative maximum step size is obtained:

$$\mu_{max} < 2 / \left( \tilde{A}_{\psi\_max} \lambda_{max} I_{i\_max}(\Lambda) \right).$$

#### Remarks

##### (R-B1): LMS and NLMS algorithms

Compared with the Gaussian case, the convergence rate of the conventional LMS and NLMS algorithms without error nonlinearity remains unchanged since  $A_\psi(\sigma_{e_g}^2, \sigma_{e_\Sigma}^2) = 1$ , though the EMSE will be increased significantly as shown below in Section 3.2.2. All the conclusions in (R-A1) and (R-A2) apply.

For general 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  $E[\mathbf{v}^T(n)\mathbf{R}_{XX}\mathbf{v}(n)]$ , as the gain  $\tilde{A}_\psi(\sigma_{e_g}^2, \sigma_{e_\Sigma}^2) = (1 - p_r)A_\psi(\sigma_{e_g}^2) + p_r A_\psi(\sigma_{e_\Sigma}^2)$  can be very small initially. This leads to nonlinear adaptation and slow convergence. Near convergence,  $\sigma_{e_g}^2$  and hence  $\tilde{A}_\psi(\sigma_{e_g}^2, \sigma_{e_\Sigma}^2)$  will become stable. The convergence is exponential and the rate for the  $i$ -th mode is approximately  $1 - \mu \tilde{A}_\psi(\infty) \lambda_i I_i(\Lambda)$ . The actual steady state value of  $\tilde{A}_\psi(\sigma_{e_g}^2(\infty), \sigma_{e_\Sigma}^2(\infty))$  will depend on the EMSE and the nonlinearity. Normally, the second term  $p_r \tilde{A}_\psi(\sigma_{e_\Sigma}^2)$  will be much smaller than the first due to the clipping property of the nonlinearity. The “asymptotic convergence rate” of the NLMS algorithm with general nonlinearity will still be faster than their LMS counterparts if the eigenvalues are unequal.

##### (R-B2): LMS and NLMS with M-nonlinearity and ATS

For the LMM/NLMM algorithms with M-nonlinearity and ATS, this degradation is again not very serious. To see this, consider the MH function where  $A_{MH}(\sigma_{e_g}^2, \sigma_{e_\Sigma}^2)$  is given by

$$\tilde{A}_{MH}(\sigma_{e_g}^2, \sigma_{e_\Sigma}^2) = (1 - p_r)A_{MH}(\sigma_{e_g}^2) + p_r A_{MH}(\sigma_{e_\Sigma}^2). \tag{50}$$

Using the  $\xi$  update in (9), the first term will approach  $(1 - p_r)A_{MH}(\sigma_{e_g}^2)$  while the second term will be close to zero if  $\sigma_{e_g}^2 \ll \sigma_{e_\Sigma}^2$  as explained at the end of Appendix C. Hence,  $A_{MH} \approx (1 - p_r)A_{MH}(\sigma_{e_g}^2)$ , which is a constant close to one if  $p_r$  is not too large.

The fastest convergence rate of these NLMS algorithms with M-nonlinearity occurs when  $\mu = 1/(\tilde{A}_\psi \lambda_{\max} I_{i,\lambda_{\max}}(\Lambda))$  and it is limited by the mode corresponding to the smallest eigenvalue. That is

$$1 - \frac{2\tilde{A}_\psi \lambda_{\min} I_{i,\lambda_{\min}}(\Lambda)}{\tilde{A}_\psi \lambda_{\max} I_{i,\lambda_{\max}}(\Lambda)} = 1 - \frac{2\lambda_{\min} I_{i,\lambda_{\min}}(\Lambda)}{\lambda_{\max} I_{i,\lambda_{\max}}(\Lambda)}. \tag{51}$$

Therefore, in additive CG noise, the maximum convergence rate of the NLMS algorithm with M-nonlinearity will also be faster than its un-normalized counterpart if the eigenvalues are unequal.

### 3.2.2 Mean Square Behavior

For convenience, we drop the argument  $(\sigma_{e_g}^2, \sigma_{e_\Sigma}^2)$  in  $\tilde{A}_\psi(\sigma_{e_g}^2, \sigma_{e_\Sigma}^2)$  and similar quantities. From (23), we have

$$\Xi(n + 1) = \Xi(n) - \mathbf{M}'_1 - \mathbf{M}'_2 + \mathbf{M}'_3, \tag{52}$$

where  $\mathbf{M}'_1 = \mu \tilde{A}_\psi \mathbf{U} \Lambda \mathbf{D}_\Lambda \mathbf{U}^T \Xi(n)$ ,  $\mathbf{M}'_2 = \mu \tilde{A}_\psi \Xi(n) \mathbf{U} \mathbf{D}_\Lambda \Lambda \mathbf{U}^T$ ,  $\mathbf{M}'_3 = \mu^2 E_{\{v, X, \eta_0\}} \left[ [\psi(e)/(\varepsilon + \mathbf{X}^T \mathbf{X})]^2 \mathbf{X} \mathbf{X}^T \right]$ . Note, the expressions for  $\mathbf{M}'_1$  and  $\mathbf{M}'_2$  are obtained by using the previous result in (48). Using (46), we have

$$\mathbf{M}'_3 = (1 - p_r) \mathbf{M}'_{3-1} + p_r \mathbf{M}'_{3-2}, \tag{53}$$

where  $\mathbf{M}'_{3-i} = \mu^2 E_{\{v, X, \eta_0\}} \left[ [\psi(e)/(\varepsilon + \mathbf{X}^T \mathbf{X})]^2 \mathbf{X} \mathbf{X}^T \right]$ , for  $i=1, 2$ , and from (27):

$$\mathbf{M}'_3 \approx 2\mu^2 \tilde{C}_\psi \mathbf{U} [\Lambda(\Xi(n) \circ \mathbf{I}(\Lambda)) \Lambda] \mathbf{U}^T + \mu^2 \tilde{B}_\psi \mathbf{U} \Lambda \mathbf{I}'(\Lambda) \mathbf{U}^T + \mu^2 \tilde{S}_\psi \mathbf{U} \tilde{\mathbf{D}}_2 \mathbf{U}^T, \tag{54}$$

where  $\tilde{C}_\psi = (1 - p_r) C_\psi(\sigma_{e_g}^2) + p_r C_\psi(\sigma_{e_\Sigma}^2)$  and

$$\tilde{S}_\psi = (1 - p_r) S_\psi(\sigma_{e_g}^2) + p_r S_\psi(\sigma_{e_\Sigma}^2)$$

$$\tilde{B}_\psi = (1 - p_r) S_\psi(\sigma_{e_g}^2) \sigma_g^2 + p_r S_\psi(\sigma_{e_\Sigma}^2) \sigma_\Sigma^2.$$

Substituting (53) into (51) and using the natural coordinate  $\Phi(n) = \mathbf{U}^T \Xi(n) \mathbf{U}$ , one gets

$$\begin{aligned} \Phi(n + 1) \approx & \Phi(n) - \mu \tilde{A}_\psi \Lambda \mathbf{D}_\Lambda \Phi(n) - \mu \tilde{A}_\psi \Phi(n) \mathbf{D}_\Lambda \Lambda \\ & + 2\tilde{C}_\psi \mu^2 [\Lambda(\Phi(n) \circ \mathbf{I}(\Lambda)) \Lambda] + \mu^2 \tilde{B}_\psi \Lambda \mathbf{I}'(\Lambda) \\ & + \mu^2 \tilde{S}_\psi \tilde{\mathbf{D}}_2(\Lambda). \end{aligned} \tag{55}$$

The  $i$ -th diagonal value of  $\Phi(n)$  is

$$\begin{aligned} \Phi_{i,i}(n + 1) \approx & \left( 1 - 2\mu \tilde{A}_\psi \lambda_i I_i(\Lambda) + 2\mu^2 \tilde{C}_\psi \lambda_i^2 I_{ii}(\Lambda) \right) \Phi_{i,i}(n) \\ & + \mu^2 \tilde{S}_\psi \sum_k \lambda_k \lambda_i I_{ki}(\Lambda) \Phi_{k,k}(n) + \mu^2 \tilde{B}_\psi \lambda_i I'_i(\Lambda). \end{aligned} \tag{56}$$

This has the same form as (28) except for the constants  $\tilde{A}_\psi$ ,  $\tilde{S}_\psi$ ,  $\tilde{B}_\psi$  and  $\tilde{C}_\psi$ . As mentioned previously, the last two terms on (55) is upper bounded at the steady state by  $\mu^2 \tilde{B}_\psi(\sigma_e^2(\infty)) \lambda_i I'_i(\Lambda)$ , where  $\tilde{B}_\psi(\sigma_e^2(\infty)) = E[\sigma_e^2(\infty) S_\psi(\sigma_e^2(\infty))]$  and the expectation, is taken over the CG noise. Consequently, we have

$$\begin{aligned} \text{EMSE}_{\text{NLMS-}\psi}(\infty) &= \text{Tr}(\Phi(\infty) \Lambda) \\ &\approx \frac{\mu \tilde{B}_\psi(\sigma_e^2(\infty))}{2\tilde{S}_\psi(\sigma_e^2(\infty))} \tilde{\phi}_{\text{NLMS-}\psi}, \end{aligned} \tag{57}$$

where  $\tilde{\phi}_{\text{NLMS-}\psi} = \tilde{S}_\psi(\sigma_e^2(\infty)) \sum_{i=1}^L \frac{\lambda_i I'_i(\Lambda)}{\tilde{A}_\psi(\sigma_e^2(\infty)) I_i(\Lambda) - \mu \tilde{C}_\psi(\sigma_e^2(\infty)) \lambda_i I_{ii}(\Lambda)}$ .

In general, (57) is a nonlinear equation in the EMSE.

#### Remarks

#### (R-B3): LMS and NLMS algorithms.

For NLMS algorithm,  $\tilde{A}_\psi = \tilde{C}_\psi = \tilde{S}_\psi = 1$ ,  $\tilde{B}_\psi = (1 - p_r) \sigma_g^2 + p_r \sigma_\Sigma^2 = \sigma_g^2 + p_r \sigma_w^2 = \sigma_{\eta_0}^2$ , and  $\tilde{B}_\psi(\sigma_e^2(\infty)) = \sigma_e^2(\infty)$ . Using the fact that  $\sigma_e^2(\infty) = \text{EMSE}_{\text{NLMS}}(\infty) + \sigma_{\eta_0}^2$ , one gets

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

Similar results are obtained for the LMS algorithm with  $I_i(\Lambda) = I'_{ii}(\Lambda) = I_{ii}(\Lambda) = 1$ . The step size bounds are the same as the Gaussian case.

#### (R-B4): LMS and NLMS algorithms with M-nonlinearities and ATS

In this case,  $\tilde{A}_\psi$ ,  $\tilde{C}_\psi$ , and  $\tilde{S}_\psi$  are nearly constant at the steady state. The step size bounds are similar to Eqs. (42) and (45) for the Gaussian noise case, except that  $A_\psi$  and  $C_\psi$  are now replaced by  $\tilde{A}_\psi$  and  $\tilde{C}_\psi$  respectively.

If  $\sigma_{e_g}^2 \ll \sigma_{e_\Sigma}^2$ , the parameter  $k_\xi$  in ATS, which is chosen as a reasonable multiple of the ‘‘impulse free’’ variance of the estimation error, can be selected to reduce the adverse effect of the impulses. Consider for example  $\tilde{A}_\psi = (1 - p_r) A_\psi(\sigma_{e_g}^2) + p_r A_\psi(\sigma_{e_\Sigma}^2)$ . As mentioned at the end of Appendix C and remarks (R-B2) above, the term  $A_\psi(\sigma_{e_\Sigma}^2)$  is usually small since the nonlinearity is designed to clip at this high noise level arising from the impulsive noise. Moreover the first term is nearly a constant. As a

result,  $\tilde{A}_\psi \approx (1 - p_r)A_\psi$ , and similarly  $\tilde{S}_\psi \approx (1 - p_r)S_\psi$ ,  $\tilde{C}_\psi = (1 - p_r)C_\psi$ , and  $\tilde{B}_\psi(\sigma_e^2(\infty)) \approx (1 - p_r)S_\psi\sigma_e^2(\infty)$ . If ATS is not employed, both  $A_\psi$  and  $C_\psi$ , and hence the MSE, will be significantly affected by the impulses. Moreover, complicated nonlinear effects will be encountered, which will generally lead to slower convergence and higher EMSE to be discussed in the sequel.

By noting that  $\sigma_e^2(\infty) = \text{EMSE}_{\text{NLMM-}\psi} + \sigma_g^2$ , (57) can be simplified to

$$\text{EMSE}_{\text{NLMS-}\psi}(\infty) \approx \frac{\frac{1}{2}\mu\sigma_g^2\tilde{\phi}_{\text{NLMS-}\psi}}{1 - \frac{1}{2}\mu\tilde{\phi}_{\text{NLMS-}\psi}}, \tag{59}$$

where  $\tilde{\phi}_{\text{NLMS-}\psi} = \tilde{S}_\psi \sum_{i=1}^L \frac{\lambda_i I'_i(\Lambda)}{A_\psi I_i(\Lambda) - \mu C_\psi \lambda_i I_{ii}(\Lambda)} \approx S_\psi \sum_{i=1}^L \frac{\lambda_i I'_i(\Lambda)}{A_\psi I_i(\Lambda) - \mu C_\psi \lambda_i I_{ii}(\Lambda)}$ , which is identical to that in Gaussian noise alone (c.f. Eq. (41)). This also holds true for the LMS-based algorithms where  $I_i(\Lambda) = I'_i(\Lambda) = I_{ii}(\Lambda) = 1$ . The step size for convergence is also identical to the Gaussian case.

For the NLMM algorithm using the MH function and ATS, the values of  $\tilde{A}_{\text{MH}}$ ,  $\tilde{S}_{\text{MH}}$  and  $\tilde{C}_{\text{MH}}$  can be computed with the help of Table 1. Assuming  $\tilde{\sigma}_g < \xi \ll \sigma_\Sigma$ , then  $\tilde{A}_{\text{MH}} \approx \tilde{S}_{\text{MH}} \approx (1 - p_r)\text{erf}\left(\frac{k_\xi}{\sqrt{2}}\right)$ ,  $\tilde{C}_{\text{MH}} \approx \tilde{A}_{\text{MH}} - (1 - p_r)\left(\frac{k_\xi}{\sqrt{2\pi}}\right)\exp\left(-\frac{k_\xi^2}{2}\right)$ . For practical values of  $k_\xi = 2.576$ , the second term of  $\tilde{C}_{\text{MH}}$  above is small and  $\tilde{\phi}_{\text{NLMS-}\psi}$  in (58) will be approximately given by

$$\tilde{\phi}_{\text{NLMS-}\psi} \approx \sum_{i=1}^L \frac{\lambda_i I'_i(\Lambda)}{I'(\Lambda) - \mu \lambda_i I_{ii}(\Lambda)}, \tag{60}$$

which is identical to that of the NLMS algorithm in Gaussian noise alone. This approximation is more accurate when the step size is small. This applies to the LMM algorithm with  $I_i(\Lambda) = I'_i(\Lambda) = I_{ii}(\Lambda) = 1$ .

In [37], the performance of the NLMS algorithm in Gaussian noise was analyzed and it is shown that  $\frac{1}{2}\mu\sigma_g^2$  serves as a useful lower bound for estimating the EMSE (misadjustment) of the NLMS algorithm. It is attractive because it does not require the knowledge of the eigenvalues or eigenvalue spread of  $\mathbf{R}_{XX}$ . A similar upper bound can be estimated empirically as well. From the above analysis for the NLMM algorithm, we see that  $\frac{1}{2}\mu\sigma_g^2$  also serves as a useful lower bound for the EMSE of the NLMM algorithm in both Gaussian and CG noises since its performance is similar to the NLMS algorithm in Gaussian noise alone in view of Eq. (60). This will be illustrated by computer simulations in the next section.

Finally, we note that the increase in EMSE of NLMS algorithm over the NLMM algorithm in CG noise environment is

$$\Delta\text{EMSE}_{\text{NLMS}}(\infty) \approx \frac{\frac{1}{2}\mu\phi_{\text{NLMS}}}{1 - \frac{1}{2}\mu\phi_{\text{NLMS}}} p_r (\sigma_\Sigma^2 - \sigma_g^2). \tag{61}$$

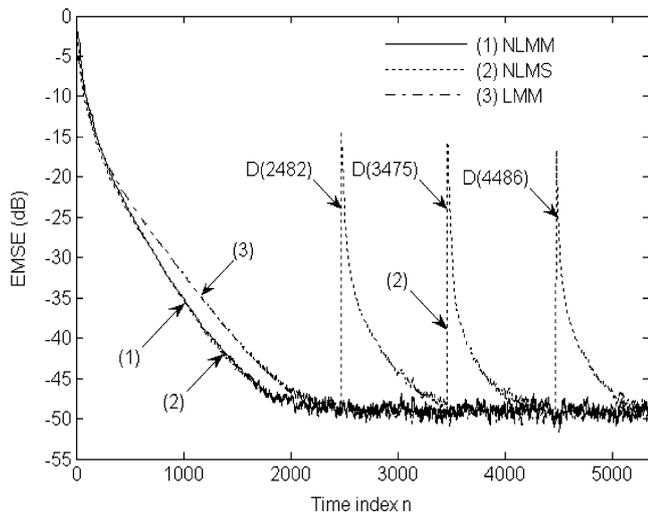
It is clear that the NLMM algorithm, which is based on robust statistics, offers a substantially lower steady state error than the NLMS algorithm in impulsive noise environment. Similar observation applies to the LMM algorithm.

### 4 Simulation Results

In this section we shall first examine the performance of the NLMM and other related algorithms and then verify the analytical results obtained in Section 3. Since the comparison of the LMM algorithm with other robust algorithms is available in [7], we will only compare the NLMM algorithm with the LMM and NLMS algorithms in our experiments. All simulations are carried out using the system identification model shown in Fig. 1 and all the learning curves are obtained by averaging the results of  $K=200$  independent runs. The unknown system to be estimated is an FIR filter with order  $L$ . Its weight vector  $\mathbf{W}^*$  is randomly generated and normalized to unit energy. The input signal  $x(n)$  is obtained from a first order AR process  $x(n) = ax(n-1) + v(n)$ , where  $0 < a < 1$  is the correlation coefficient.  $v(n)$  is an additive white Gaussian noise (AWGN) sequence with zero mean and variance  $\sigma_v^2$  and  $\sigma_v^2 = 1/(1 - a^2)$ . The additive noise is either the Gaussian noise  $\eta_g(n)$  with zero mean and variance  $\sigma_g^2$ , or the CG noise sequence  $\eta_0(n)$  generated from (12) with a probability  $P(b(n) = 1) = p_r$ .

#### Experiment A: Convergence performance in Gaussian and CG noise

In this experiment, the convergence performance and the robustness of the NLMM algorithm to impulsive noise are evaluated. The system order is  $L=8$  and the correlation coefficient of the input AR process is  $a=0.9$ . The impulse occurrence probability and the impulsive characteristic ratio are respectively  $p_r=0.01$  and  $r_{im}=300$ . For illustration purpose, the impulsive noise is applied to the system after time instant  $n=2,000$ . From  $n=1$  to  $n=1,999$ , the additive noise is  $\eta_g(n)$  with  $\sigma_g^2 = 10^{-4}$ . From  $n=2,000$  onwards, a CG noise  $\eta_0(n)$  generated by (12) is applied. The step size of the NLMM, NLMS and LMM algorithms are set to  $\mu_{\text{NLMM}} = \mu_{\text{NLMS}} = 0.1$  and  $\mu_{\text{LMM}}=0.025$ , which enables all algorithms to reach a similar steady state EMSE. The small positive constant  $\epsilon$  used to prevent division by zero in (2) and (10) for the NLMS and NLMM algorithms is set to 0.0001. The threshold parameter  $\xi$  of the M-estimate function in the NLMM and LMM algorithms is calculated from (9) with  $\hat{\sigma}_e^2(n)$  being estimated from (7). The forgetting factor  $\lambda_\sigma$  in (7) is 0.95. The window length  $N_w$  for the NLMM and the LMM algorithms is chosen to be 9. For better visualization, the locations of the impulses



**Figure 3** The convergence performance and the robustness of (1) LMM, (2) NLMM and (3) NLMS algorithms to impulses in desired signal (MSE vs.  $n$ ).

are fixed whereas their amplitudes are varied according to  $\eta_w(n)$ . This is realized by generating one fixed Bernoulli sequence  $\bar{b}(n)$  with  $p_r=0.01$  and using it in all of the independent runs. Consequently, the locations of the impulses are fixed at  $n=2,482$ ,  $n=3,475$  and  $n=4,486$  so as to visualize more clearly the impact of the impulsive noise on the desired signal. Figure 3 depicts the performance of all the algorithms, which shows they can reach a similar steady state MSE value. The NLMM and NLMS algorithms have almost identical initial convergence performance. It can be also seen that the NLMM and LMM algorithms are very robust to the impulses in the desired signal. In contrast, the performance of the NLMS algorithm is severely deteriorated by these impulses.

Simulations for  $p_r$  larger than 0.01 were also conducted and similar results are obtained. In general, the performance of the algorithms will be degraded gradually as  $p_r$  increases. More simulation results concerning the effects of using different parameter values of SNR,  $\mu_{NLMM}$ ,  $N_w$ ,  $k_\xi$  are available in [41]. The results show that the NLMM algorithm has improved robustness to impulses and is not

**Table 2** List of parameters in experiment B.

Exps.	$L$	$a$	$\mu$	$\sigma_g^2$	$r_{im}$	$p_r$	Fig. and curve index		
							Mean	Mean square	$e^2(n), \hat{\sigma}_e^2$
(1). NLMS with CG noise	16	0	0.2	$10^{-6}$	50	0.005	4 (a), (1)	4 (b), (1)	N/A
	16	0.3	0.15	$10^{-5}$	100	0.01	4 (a), (2)	4 (b), (2)	N/A
	16	0.6	0.1	$10^{-4}$	200	0.015	4 (a), (3)	4 (b), (3)	N/A
	16	0.9	0.05	$10^{-3}$	400	0.02	4 (a), (4)	4 (b), (4)	N/A
(2). NLMM with CG noise	8	0	0.2	$10^{-6}$	50	0.02	5 (a), (1)	5 (b), (1)	Omitted
	8	0	0.1	$10^{-6}$	50	0.02	5 (a), (2)	5 (b), (2)	5 (c), (2)
	8	0	0.05	$10^{-6}$	50	0.02	5 (a), (3)	5 (b), (3)	Omitted
	8	0	0.02	$10^{-3}$	400	0.005	5 (a), (4)	5 (b), (4)	Omitted
	8	0	0.01	$10^{-3}$	400	0.005	5 (a), (5)	5 (b), (5)	5 (c), (5)
	8	0	0.007	$10^{-3}$	400	0.005	5 (a), (6)	5 (b), (6)	Omitted
	24	0.9	0.3	$10^{-6}$	50	0.02	6 (a), (1)	6 (b), (1)	6 (c), (1)
	24	0.9	0.15	$10^{-6}$	50	0.02	6 (a), (2)	6 (b), (2)	Omitted
	24	0.9	0.1	$10^{-6}$	50	0.02	6 (a), (3)	6 (b), (3)	Omitted
	24	0.9	0.08	$10^{-3}$	400	0.005	6 (a), (4)	6 (b), (4)	Omitted
(3). NLMM with Gaussian noise	8	0	0.2	$10^{-6}$	N/A	N/A	7 (a), (1)	7 (b), (1)	Omitted
	8	0.3	0.15	$10^{-5}$	N/A	N/A	7 (a), (2)	7 (b), (2)	7 (c), (2)
	8	0.6	0.1	$10^{-4}$	N/A	N/A	7 (a), (3)	7 (b), (3)	Omitted
	8	0.9	0.05	$10^{-3}$	N/A	N/A	7 (a), (4)	7 (b), (4)	7 (c), (4)
(4). LMM with CG noise	24	0	0.01	$10^{-6}$	50	0.02	8 (a), (1)	8 (b), (1)	8 (c), (1)
	24	0.3	0.008	$10^{-5}$	100	0.015	8 (a), (2)	8 (b), (2)	Omitted
	24	0.6	0.006	$10^{-4}$	200	0.01	8 (a), (3)	8 (b), (3)	Omitted
	24	0.9	0.005	$10^{-3}$	400	0.005	8 (a), (4)	8 (b), (4)	8 (c), (4)

too sensitive to these parameters if they are reasonably chosen as suggested.

**Experiment B: Verification of convergence performance analysis**

In this experiment, the theoretical analysis presented in Section 3 will be verified through extensive simulations. For the mean convergence, the norm of the mean square weight-error vector is used as the performance measure:

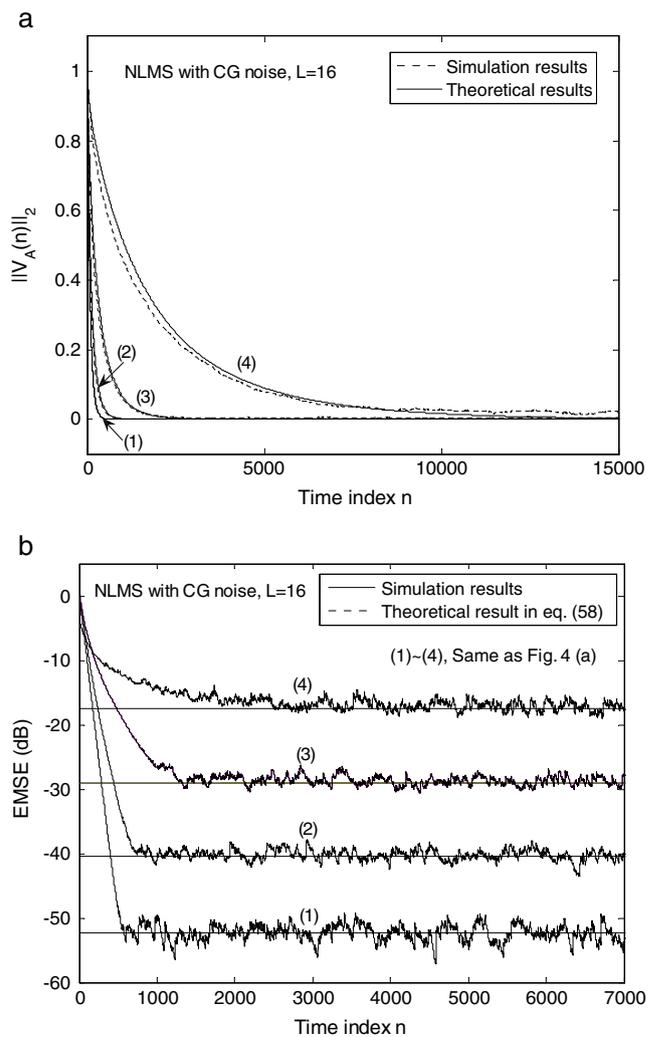
$$\|v_A(n)\|_2 = \sqrt{\sum_{i=1}^L \left[ \frac{1}{K} \sum_{j=1}^K v_i^{(j)}(n) \right]^2}, \quad i = 1, \dots, L,$$

$$j = 1, \dots, K,$$

where  $v_i^{(j)}(n)$  is the  $i$ -th component of the weight-error vector  $v(n)$  at time  $n$  in the  $j$ -th independent run. For the mean square convergence results,  $EMSE(n) = \text{Tr}(\Phi(n)\Lambda)$  is used as the performance measure. The values of the special integral functions appearing in the analytical results, i. e.,  $I_i(\Lambda)$ ,  $I'_i(\Lambda)$ , and  $I_{ii}(\Lambda)$ , are evaluated numerically using the method introduced in [45]. Unlike experiment A, the impulsive noise sequence used in these algorithms is applied to the tested algorithms throughout the whole adapting process. The locations of impulses in the desired signal are not fixed for each independent run. Different system order  $L$ , input correlation factor  $a$ , algorithm step size  $\mu$  and other key parameters  $\sigma_g^2$ ,  $r_{im}$ , and  $p_r$  are employed in different experiments. For succinct description, their values are summarized in Table 2. The following four experiments are conducted:

Ex.1 NLMS algorithm with CG noise. The theoretical results are calculated from (49), (56) and (58). Figure 4a shows that for mean convergence there is a good agreement between theoretical and experimental results. It can be seen from Fig. 4b that the EMSE as given in (58) is also close to the true EMSE, which shows that the performance of the NLMS is substantially affected by CG noise. The results concerning NLMS algorithm with Gaussian noise has been detailed in [37] so that they are omitted here.

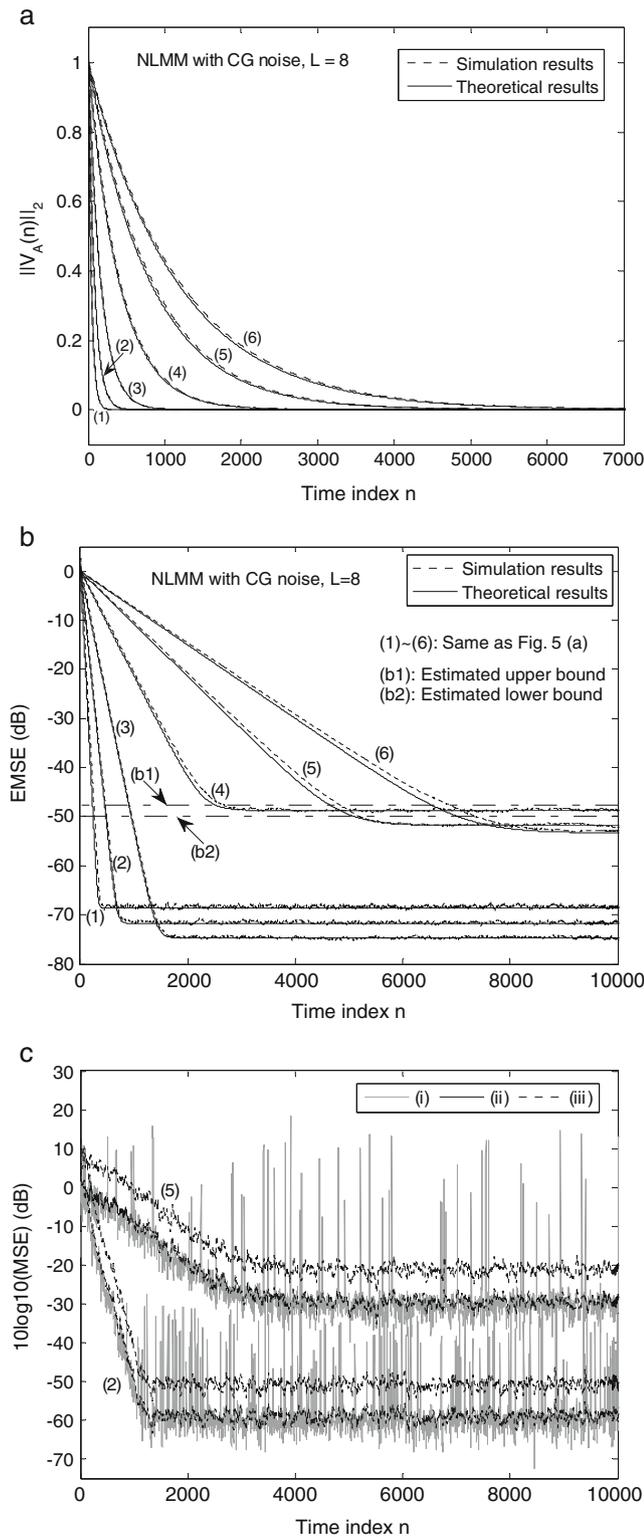
Ex.2 NLMM algorithm with CG and Gaussian noises. The theoretical results of the former are calculated from (49), (56) and (60) while those for Gaussian noise are calculated from (20), (29) and (44). All theoretical and experimental results are depicted in Figs. 5, 6 and 7 respectively and they are found to match each other very well. To evaluate the effectiveness and robustness of the ATS proposed in (7), (9), the “impulse-free” variance  $\hat{\sigma}_e^2(n)$  estimated from (7),  $k_{\xi}^2 \hat{\sigma}_e^2(n)$ , and  $\sigma_e^2(n)$  for one independent run are plotted in sub-figures (c) at Figs. 5, 6, 7 and 8. It can be seen that the estimated “impulse-free” variance is able to follow the true value



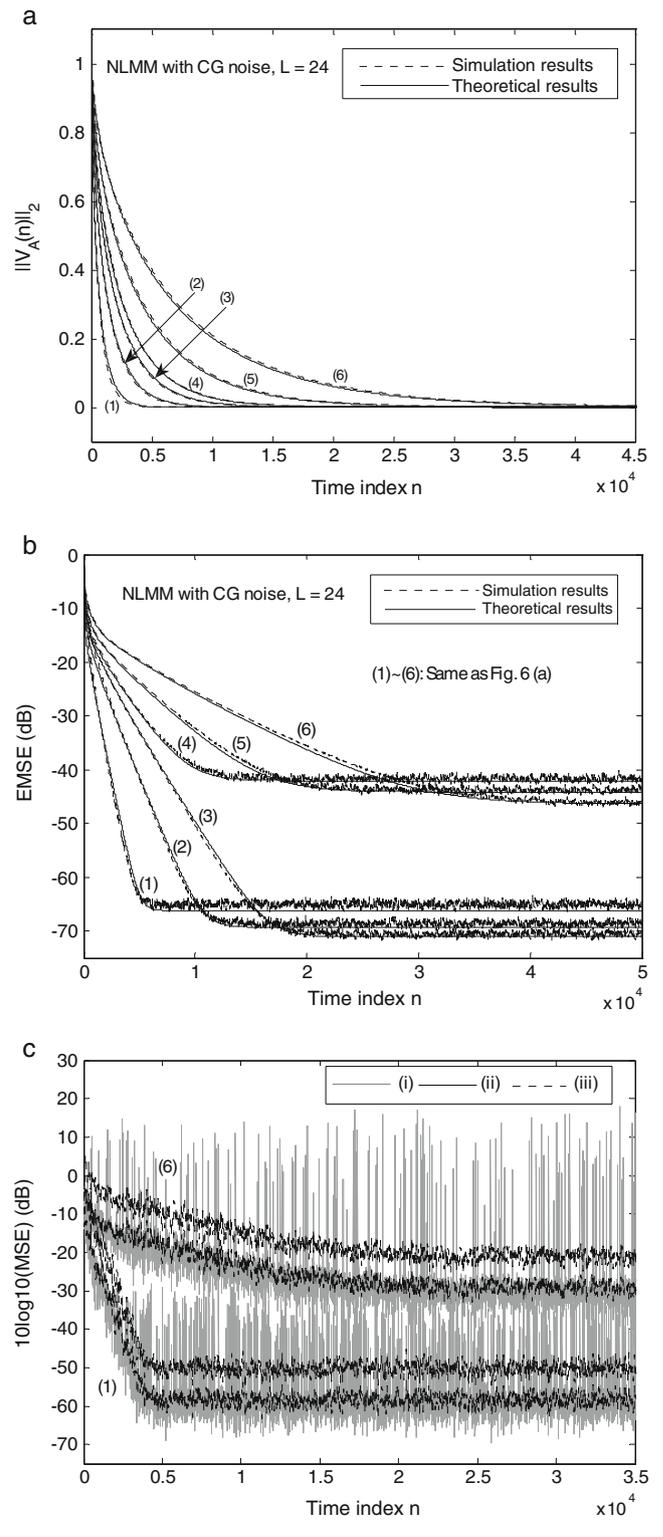
**Figure 4** The mean (a) and mean square (b) convergence performance of the NLMS algorithm with CG noise. Learning curves: (1)  $a = 0, \mu = 0.2, \sigma_g^2 = 10^{-6}, r_{im} = 50, p_r = 0.005$ ; (2)  $a = 0.3, \mu = 0.15, \sigma_g^2 = 10^{-5}, r_{im} = 100, p_r = 0.01$ ; (3)  $a = 0.6, \mu = 0.1, \sigma_g^2 = 10^{-4}, r_{im} = 200, p_r = 0.015$ ; (4)  $a = 0.9, \mu = 0.05, \sigma_g^2 = 10^{-3}, r_{im} = 400, p_r = 0.02$ .

and impulses with large amplitudes can be effectively detected. As pointed out at the end of Section 3, an estimate of the lower bound of the EMSE for the NLMM algorithm is  $\frac{1}{2} \mu \sigma_g^2$ . To account for the approximation error in this formula, we also empirically estimated an upper bound  $CF \cdot \frac{1}{2} \mu \sigma_g^2$  from extensive simulation results by means of a constant correction factor  $CF$ , which was found to be 1.7. Due to space limitation, we only plot these bounds in Fig. 5b curve (4) and Fig. 7b curve (3). It can be seen that these estimates give satisfactory bounds for the EMSE.

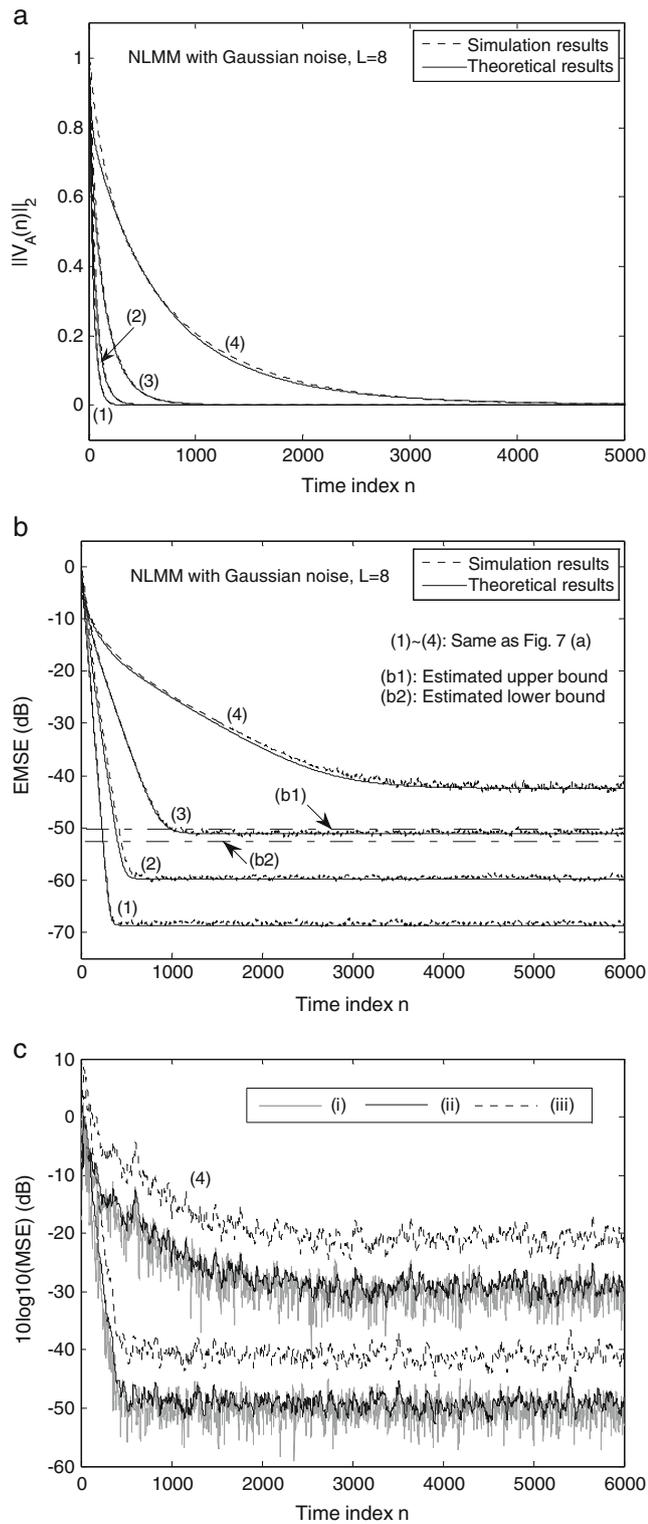
Ex.3 LMM algorithm with CG noise, which is a special case of Ex. 1 with all special integrals equal to one. Figure 8 illustrates the good performance of the algorithm and an agreement between the theoretical and simulation results.



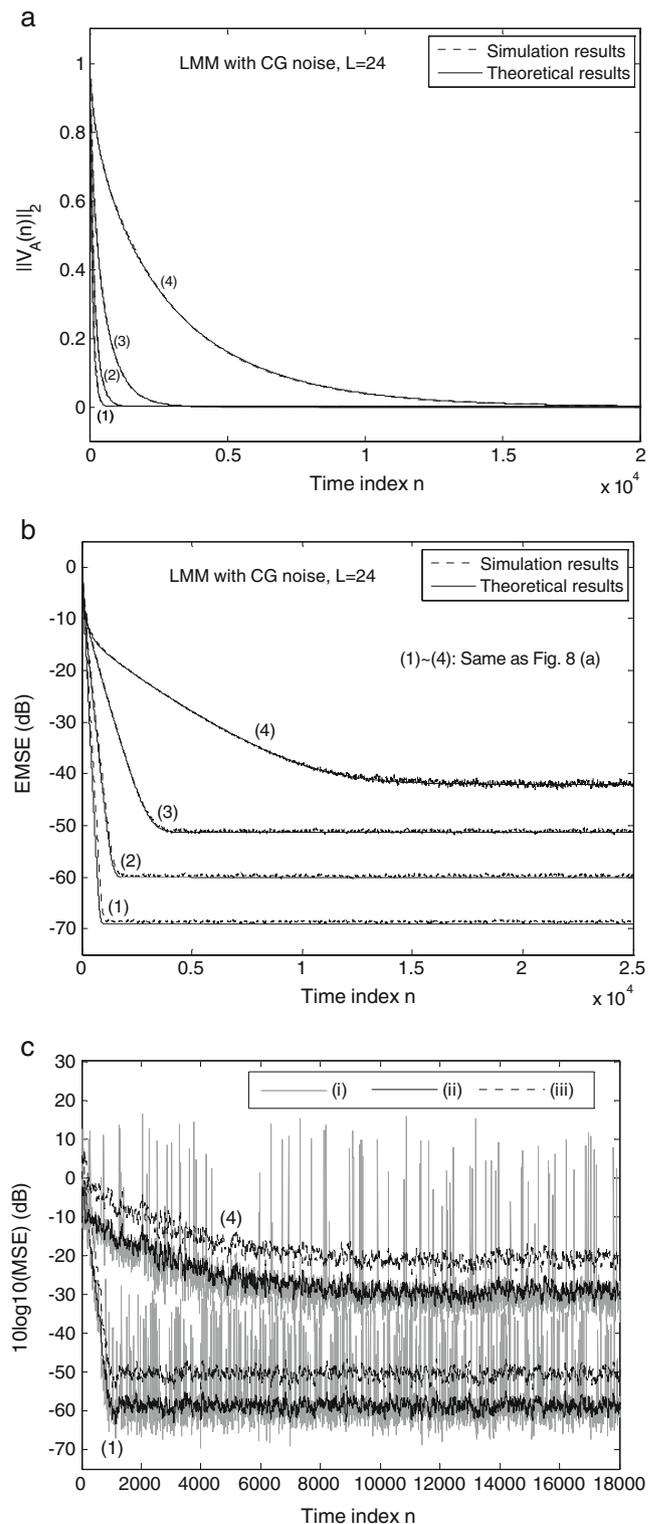
**Figure 5** The mean (a) and mean square (b) convergence performance of the NLMM algorithm with CG noise; c Illustration of (i)  $\sigma_e^2(n)$ , (ii)  $\hat{\sigma}_e^2(n)$  and (iii)  $k_e^2 \hat{\sigma}_e^2(n)$ . Learning curves: (1)  $a = 0, \mu = 0.2, \sigma_g^2 = 10^{-6}, r_{im} = 50, p_r = 0.02$ ; (2)  $a = 0, \mu = 0.01, \sigma_g^2 = 10^{-6}, r_{im} = 50, p_r = 0.02$ ; (3)  $a = 0, \mu = 0.05, \sigma_g^2 = 10^{-6}, r_{im} = 50, p_r = 0.02$ ; (4)  $a = 0, \mu = 0.02, \sigma_g^2 = 10^{-3}, r_{im} = 400, p_r = 0.005$ ; (5)  $a = 0, \mu = 0.01, \sigma_g^2 = 10^{-3}, r_{im} = 400, p_r = 0.005$ ; (6)  $a = 0, \mu = 0.07, \sigma_g^2 = 10^{-3}, r_{im} = 400, p_r = 0.005$ .



**Figure 6** The mean (a) and mean square (b) convergence performance of the NLMM algorithm with CG noise; c Illustration of (i)  $\sigma_e^2(n)$ , (ii)  $\hat{\sigma}_e^2(n)$  and (iii)  $k_e^2 \hat{\sigma}_e^2(n)$ . Learning curves: (1)  $a = 0.9, \mu = 0.3, \sigma_g^2 = 10^{-6}, r_{im} = 50, p_r = 0.02$ ; (2)  $a = 0.9, \mu = 0.15, \sigma_g^2 = 10^{-6}, r_{im} = 50, p_r = 0.02$ ; (3)  $a = 0.9, \mu = 0.1, \sigma_g^2 = 10^{-6}, r_{im} = 50, p_r = 0.02$ ; (4)  $a = 0.9, \mu = 0.08, \sigma_g^2 = 10^{-3}, r_{im} = 400, p_r = 0.005$ ; (5)  $a = 0.9, \mu = 0.05, \sigma_g^2 = 10^{-3}, r_{im} = 400, p_r = 0.005$ ; (6)  $a = 0.9, \mu = 0.03, \sigma_g^2 = 10^{-3}, r_{im} = 400, p_r = 0.005$ .



**Figure 7** The mean (a) and mean square (b) convergence performance of the NLMM algorithm with Gaussian noise; c Illustration of (i)  $\sigma_e^2(n)$ , (ii)  $\hat{\sigma}_e^2(n)$  and (iii)  $k_g^2 \hat{\sigma}_e^2(n)$ . Learning curves: (1)  $a = 0, \mu = 0.02, \sigma_g^2 = 10^{-6}$ ; (2)  $a = 0.3, \mu = 0.15, \sigma_g^2 = 10^{-5}$ ; (3)  $a = 0.6, \mu = 0.01, \sigma_g^2 = 10^{-4}$ ; (4)  $a = 0.9, \mu = 0.05, \sigma_g^2 = 10^{-3}$ .



**Figure 8** The mean (a) and mean square (b) convergence performance of the LMM algorithm with CG noise; c Illustration of (i)  $\sigma_e^2(n)$ , (ii)  $\hat{\sigma}_e^2(n)$  and (iii)  $k_g^2 \hat{\sigma}_e^2(n)$ . Learning curves: (1)  $a = 0, \mu = 0.01, \sigma_g^2 = 10^{-6}, r_{im} = 50, p_r = 0.02$ ; (2)  $a = 0.3, \mu = 0.008, \sigma_g^2 = 10^{-5}, r_{im} = 100, p_r = 0.015$ ; (3)  $a = 0.6, \mu = 0.006, \sigma_g^2 = 10^{-4}, r_{im} = 200, p_r = 0.01$ ; (4)  $a = 0.9, \mu = 0.005, \sigma_g^2 = 10^{-3}, r_{im} = 400, p_r = 0.005$ .

### 5 Conclusions

A new convergence performance analysis of the LMS and NLMS algorithms with error nonlinearity in Gaussian and CG noises is presented. The approach relies on the use of the Price theorem and its extension to signals with mixture Gaussian distributions, and the use of generalized Abelian integral functions in evaluating the mathematical expectation involved. New formulas for the EMSE, stability bound and difference equations describing both the transient and the steady state convergence behaviors of the algorithms are obtained. Simulation results show good agreement with the theoretical results.

### Appendix

#### Appendix A: Evaluation of $H_1$

We now extend the approach in [17, 18] and [12] to evaluate  $H_1$ . Using the independence assumption 3, we further have  $H_1 = E_{\{v\}}[H]$ , where  $H = E_{\{X, \eta_g\}}[\psi(e)X / (\varepsilon + X^T X) | v]$ . As  $\eta_g(n)$  and  $x(n)$  are also assumed to be statistically independent, and  $X$  are jointly Gaussian with autocorrelation matrix  $R_{XX}$ , one gets

$$H = C_R \int\int_{L+1 \text{ fold}} \frac{\psi(e)X}{\varepsilon + X^T X} \exp(-\frac{1}{2}X^T R_{XX}^{-1} X) f_{\eta_g}(\eta_g) d\eta_g dX, \tag{A-1}$$

where  $C_R = (2\pi)^{-L/2} |R_{XX}|^{-1/2}$  and  $f_{\eta_g}(\eta_g)$  is the PDF of the Gaussian noise  $\eta_g$ .  $|\cdot|$  denotes the determinant of a matrix. Similar to [17] and [18], let us consider the integral

$$F(\beta) = C_R \int\int_{L+1 \text{ fold}} \frac{\psi(e)X \exp(-\beta(\varepsilon + X^T X))}{\varepsilon + X^T X} \cdot \exp(-\frac{1}{2}X^T R_{XX}^{-1} X) f_{\eta_g}(\eta_g) d\eta_g dX. \tag{A-2}$$

It can be seen that  $H=F(0)$ . Differentiating (A-2) with respect to  $\beta$ , one gets

$$\frac{dF(\beta)}{d\beta} = -\exp(-\beta\varepsilon) C_R \int\int_{L+1 \text{ fold}} (\psi(e)X) \exp(-\frac{1}{2}X^T B^{-1}(n)X) f_{\eta_g}(\eta_g) d\eta_g dX, \tag{A-3}$$

where  $B(n) = (2\beta I + R_{XX}^{-1})^{-1}$ . For notation convenience, we shall simply write  $B$  for  $B(n)$ . To evaluate the integral, we use the eigenvalue decomposition of  $R_{XX} = U\Lambda U^T$ , where  $\Lambda$  is a diagonal matrix whose elements are the eigenvalues of  $R_{XX}$  and  $U$  is a unitary matrix. The matrix  $B^{-1}$  can be written as

$$B^{-1} = U(2\beta I + \Lambda^{-1})U^T = UD^{-1}U^T, \tag{A-4}$$

where  $D$  is a diagonal matrix with the  $i$ -th diagonal entry given by  $d_{ii}(n) = (2\beta + \lambda_i^{-1})^{-1}$ . Noting that the determinants

of  $U$  and  $D$  are respectively 1 and  $|D|^{-1/2} = \prod_{i=1}^L (2\beta + \lambda_i^{-1})^{1/2}$ , one can rewrite (A-3) as follows

$$\frac{dF(\beta)}{d\beta} = -\gamma(\beta) E_{\{X, \eta_g\}}[\psi(e)X | v] |_{E[XX^T]=B} = -\gamma(\beta)L_1, \tag{A-5}$$

where  $C_B = (2\pi)^{-L/2} |B|^{-1/2}$ ,  $\gamma(\beta) = \exp(-\beta\varepsilon) \prod_{i=1}^L (2\beta\lambda_i + 1)^{-1/2}$ , and  $L_1 = E_{\{X, \eta_g\}}[\psi(e)X | v] |_{E[XX^T]=B}$  is the expectation of  $\psi(e)X$  conditioned on  $v$  when  $x_i, x_j \in X$  are jointly Gaussian with correlation matrix  $B$ . We shall evaluate  $L_1$  by considering its  $i$ -th element as follows:

$$L_{1,i} = E_{\{X, \eta_g\}}[\psi(e)x_i | v] |_{E[XX^T]=B}. \tag{A-6}$$

Using the Price's theorem, we have

$$\begin{aligned} \frac{\partial L_{1,i}}{\partial r_{x_i e}} &= E_{\{X, \eta_g\}} \left[ \frac{d\psi(e)}{de} | v \right] |_{E[XX^T]=B} \\ &= \frac{1}{\sqrt{2\pi}\sigma_e} \int_{-\infty}^{\infty} \psi'(e) \exp\left(-\frac{e^2}{2\sigma_e^2}\right) de = \overline{\psi'}(\sigma_e^2), \end{aligned} \tag{A-7}$$

where  $\overline{\psi'}(\sigma_e^2)$  is the average of  $\psi'(e)$ , and  $e = X^T v + \eta_g$ , which is Gaussian distributed with zero mean and variance  $\sigma_e^2(v) = E[v^T X X^T v | v] |_{E[XX^T]=B} + \sigma_g^2 = v^T B v + \sigma_g^2$ . To simplify notation, we shall write  $\overline{\psi'}(\sigma_e^2)$  as  $A_\psi(\sigma_e^2)$ . Integrating (A-7) with respect to  $r_{x_i e}$ , one gets

$$L_{1,i} = A_\psi(\sigma_e^2) r_{x_i e}, \tag{A-8}$$

where the constant of integration is zero because  $r_{x_i e}$  is equal to zero when  $x_i$  and  $e$  are uncorrelated. Since

$$\begin{aligned} r_{x_i e} &= E[x_i e | v] |_{E[XX^T]=B} \\ &= E[x_i (X^T v + \eta_g) | v] |_{E[XX^T]=B} = B_i v, \end{aligned} \tag{A-9}$$

where  $B_i$  is the  $i$ -th row of  $B$ , we have

$$L_1 = A_\psi(\sigma_e^2) B v(n). \tag{A-10}$$

Substituting (A-10) into (A-5) and integrating yield

$$\begin{aligned} F(\beta) &= (-)^{\beta} A_\psi(\sigma_e^2) \gamma(\beta) B d\beta v(n) \\ &\approx A_\psi(\overline{\sigma_e^2}) I(\beta) v(n), \end{aligned} \tag{A-11}$$

where  $I(\beta) = U(-)^{\beta} \gamma(\beta) D(n) d\beta U^T$  and  $d_{ii}(n) = \lambda_i / (2\beta\lambda_i + 1)$ . The constant of integration is equal to zero because of the boundary condition  $F(\infty)=0$ . Here, we have assumed that  $\overline{\psi'}(\sigma_e^2)$  depends weakly on  $\beta$  and is taken outside of the integral using mean value theorem with some appropriate mean value  $A_\psi(\overline{\sigma_e^2})$ . Note, the dependence of  $A_\psi(\sigma_e^2) = \overline{\psi'}(\sigma_e^2)$  on  $\beta$  is mainly through  $\sigma_e^2 = v^T B v + \sigma_g^2$ . From (A-4), we can see that as  $\beta$  varies from 0 to  $\infty$ ,  $\sigma_e^2$  will decrease from  $\sigma_e^2(n) = v^T R_{XX} v + \sigma_g^2$  to  $\sigma_g^2$ . Here, we shall use the mean value of the upper bound:  $\sigma_e^2(n) = E[v^T R_{XX} v] + \sigma_g^2$ , which is the mean MSE at time  $n$  for  $\sigma_e^2$ . This is also a good approximation at the steady

state of the algorithm where  $\sigma_e^2 = \mathbf{v}^T \mathbf{B} \mathbf{v} + \sigma_g^2 \approx \sigma_g^2$ . This is also commonly used as in [34, 35]. For the effect of the nonlinearity  $A_\psi(\sigma_e^2)$ , first we note from (A-7) that it can be rewritten as  $A_\psi(\sigma_e^2) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} \psi'(\sigma_e u) \exp\left(-\frac{u^2}{2}\right) du$ . For most practical nonlinearities, it is shown in Appendix C that  $\psi(e)$  can be written as  $eq(e/\xi)$  with threshold parameter  $\xi$ . Moreover, it can be shown in Appendix C that if the ATS in (7)–(9) is used to update  $\xi$  for these nonlinearities (collectively called the M-nonlinearity) and the estimated error variance  $\hat{\sigma}_e^2$  is close to the “impulse-free” variance, then  $A_\psi(\sigma_e^2)$  is approximately independent of  $\sigma_e^2$  and hence of  $\beta$ .

For general nonlinearity,  $\psi(e)$  is usually equal to one when  $|e|$  is smaller than  $\xi$ , and gradually decreases to zero. For sufficiently large  $\xi$ , the main contribution is due to the first term in (C-5) because of the exponential decay of the Gaussian PDF. Hence,  $A_\psi(\sigma_e^2) \approx \frac{2}{\sqrt{2\pi}} \int_0^{\xi/\sigma_e} \exp\left(-\frac{u^2}{2}\right) du$ . The derivative of  $A_\psi(\sigma_e^2)$  with respect to  $\sigma_e^2$  is approximately  $-\frac{2\xi}{\sqrt{2\pi}\sigma_e^2} \exp\left(-\frac{\xi^2}{2\sigma_e^2}\right)$ . For sufficiently large  $\xi$ , the exponential factor decreases faster than the other factor and the variations of  $A_\psi(\sigma_e^2)$  with respect to  $\sigma_e^2$  is small. This also justifies the approximation used above. To efficiently suppress outliers,  $\xi$  however cannot be very large so that it is important to adapt it with respect to  $\sigma_e^2$ , which will help to reduce the variation of  $A_\psi(\sigma_e^2)$  with respect to  $\sigma_e^2$ . These are also the cases which are of practical interests. In conclusion, (A-11) is a good approximation for commonly used M-estimate functions and practical nonlinearities with ATS or at the steady state of the algorithm.

To evaluate the term  $\mathbf{I}(0)$  and hence  $\mathbf{H} = \mathbf{F}(0) = A_\psi(\sigma_e^2) \mathbf{I}(0) \mathbf{v}(n)$ , we note from (A-4) that  $\mathbf{B} = \mathbf{U} \mathbf{D}(n) \mathbf{U}^T$  and thus

$$\mathbf{I}(0) = \mathbf{U} \left( -\int^0 \gamma(\beta) \mathbf{D}(n) d\beta \right) \mathbf{U}^T = \mathbf{U} \mathbf{\Lambda} \mathbf{D}_\Lambda \mathbf{U}^T. \tag{A-12}$$

where  $\mathbf{D}_\Lambda = \text{diag}(I_1(\Lambda), \dots, I_L(\Lambda))$  and

$$I_i(\Lambda) = \int_0^\infty \exp(-\beta \varepsilon) \left[ \prod_{k=1}^L (2\beta \lambda_k + 1)^{-1/2} \right] (2\beta \lambda_i + 1)^{-1} d\beta, \tag{A-13}$$

$i = 1, 2, \dots, L$ . Finally, we have the following desired result

$$\mathbf{H}_1 = E[\mathbf{H}] = E[\mathbf{F}(0)] \approx A_\psi(\sigma_e^2) \mathbf{U} \mathbf{\Lambda} \mathbf{D}_\Lambda \mathbf{U}^T E[\mathbf{v}(n)]. \tag{A-14}$$

For the LMS-like algorithms,  $\mathbf{B}$  will be equal to  $\mathbf{R}_{XX}$ . Accordingly, (A-14) will be modified by replacing  $\mathbf{D}_\Lambda$  with the identity matrix and  $\sigma_e^2(\mathbf{v})$  by  $\mathbf{v}^T \mathbf{R}_{XX} \mathbf{v} + \sigma_g^2$ .

### Appendix B: Evaluation of $s_3$ and $\mathbf{M}_3$

Firstly, we note that  $\mathbf{M}_3 = \mu^2 E_{\{\mathbf{v}\}}[s_3]$ , where  $s_3 = E_{\{X, \eta_g\}} \left[ \psi^2(e) \mathbf{X} \mathbf{X}^T / (\varepsilon + \mathbf{X}^T \mathbf{X})^2 | \mathbf{v} \right]$ . As in Appendix A, we can write  $s_3$  as:

$$s_3 = C_R \iint_{L+1\text{fold}} \frac{\psi^2(e) \mathbf{X} \mathbf{X}^T}{(\varepsilon + \mathbf{X}^T \mathbf{X})^2} \exp\left(-\frac{1}{2} \mathbf{X}^T \mathbf{R}_{XX}^{-1} \mathbf{X}\right) f_{\eta_g}(\eta_g) d\eta_g d\mathbf{X}. \tag{B-1}$$

Let us define

$$\overline{\mathbf{F}}(\beta) = C_R \iint_{L+1\text{fold}} \frac{\psi^2(e) \mathbf{X} \mathbf{X}^T \exp(-\beta(\varepsilon + \mathbf{X}^T \mathbf{X}))}{(\varepsilon + \mathbf{X}^T \mathbf{X})^2} \times \exp\left(-\frac{1}{2} \mathbf{X}^T \mathbf{R}_{XX}^{-1} \mathbf{X}\right) f_{\eta_g}(\eta_g) d\eta_g d\mathbf{X}. \tag{B-2}$$

Comparing (B-2) with (B-1), it can be seen that  $s_3 = \overline{\mathbf{F}}(0)$ . To evaluate  $\overline{\mathbf{F}}(\beta)$ , differentiating (B-2) twice with respect to  $\beta$ , one gets

$$\begin{aligned} d^2 \overline{\mathbf{F}}(\beta) / d\beta^2 &= C_R \exp(-\beta \varepsilon) \iint_{L+1\text{fold}} \psi^2(e) \mathbf{X} \mathbf{X}^T \exp\left(-\frac{1}{2} \mathbf{X}^T \mathbf{B}^{-1} \mathbf{X}\right) f_{\eta_g}(\eta_g) d\eta_g d\mathbf{X} \\ &= \gamma(\beta) \left[ C_B \iint_{L+1\text{fold}} \psi^2(e) \mathbf{X} \mathbf{X}^T \exp\left(-\frac{1}{2} \mathbf{X}^T \mathbf{B}^{-1} \mathbf{X}\right) f_{\eta_g}(\eta_g) d\eta_g d\mathbf{X} \right] = \gamma(\beta) \mathbf{L}_3, \end{aligned} \tag{B-3}$$

where  $\mathbf{L}_3 = E_{\{X, \eta_g\}} \left[ \psi^2(e) \mathbf{X} \mathbf{X}^T | \mathbf{v} \right] \Big|_{E[\mathbf{X} \mathbf{X}^T] = \mathbf{B}}$ ,  $\gamma(\beta)$ ,  $C_B$ , and the correlation matrix  $\mathbf{B}$  have been defined in Appendix A. Let's evaluate the  $(i, j)$ -th element of  $\mathbf{L}_3$  as follows:

$$\mathbf{L}_{3,ij} = E_{\{X, \eta_g\}} \left[ \psi^2(e) x_i x_j | \mathbf{v} \right] \Big|_{\mathbf{B}}. \tag{B-4}$$

Using Price's theorem, we have

$$\begin{aligned} \frac{\partial \mathbf{L}_{3,ij}}{\partial r_{x_i x_j}} &= E_{\{X, \eta_g\}} \left[ \psi^2(e) | \mathbf{v} \right] \Big|_{\mathbf{B}} \\ &= \frac{1}{\sqrt{2\pi}\sigma_e} \int_{-\infty}^{\infty} \psi^2(e) \exp\left(-\frac{e^2}{2\sigma_e^2}\right) de = \overline{\psi^2}(\sigma_e^2). \end{aligned} \tag{B-5}$$

For notational simplicity, we shall write  $\overline{\psi^2}(\sigma_e^2)$  as  $B_\psi(\sigma_e^2)$ . Integrating with respect to  $r_{x_i x_j}$  gives

$$\mathbf{L}_{3,ij} = B_\psi(\sigma_e^2) r_{x_i x_j} + c_{ij}, \tag{B-6}$$

where  $c_{ij} = E \left[ \psi^2(e) x_i x_j \right] \Big|_{r_{x_i x_j} = 0}$  is the integration constant. Using Price's theorem again, we have

$$\frac{\partial c_{ij}}{\partial r_{x_i x_j}} = E \left[ \frac{d\psi^2(e)}{de} x_j \right] \Big|_{r_{x_i x_j} = 0} = E \left[ \frac{d^2 \psi^2(e)}{de^2} \right] r_{x_i x_j} = 2C_\psi(\sigma_e^2) r_{x_i x_j}. \tag{B-7}$$

(From Price’s theorem,  $\frac{1}{2}E\left[\frac{d^2\psi^2(e)}{de^2}\right] = \frac{d}{d\sigma_e^2}E[\psi^2(e)]$ , which is written as  $C_\psi(\sigma_e^2)$  here) Integrating again, one gets

$$c_{ij} = 2C_\psi(\sigma_e^2)r_{xje}r_{xie}. \tag{B-8}$$

Combining (B-6) and (B-8), we have

$$\mathbf{L}_3 = B_\psi(\sigma_e^2)\mathbf{B} + 2C_\psi(\sigma_e^2)\mathbf{B}\mathbf{v}\mathbf{v}^T\mathbf{B}. \tag{B-9}$$

Substituting (B-9) into (B-3) gives

$$\frac{d^2\bar{\mathbf{F}}(\beta)}{d\beta^2} = \gamma(\beta)B_\psi(\sigma_e^2)\mathbf{B} + 2\gamma(\beta)C_\psi(\sigma_e^2)\mathbf{B}\mathbf{v}\mathbf{v}^T\mathbf{B}. \tag{B-10}$$

Integrating (B-10) with respect to  $\beta$  yields

$$s_3 = \int_0^\infty \int_{\beta_1}^\infty \gamma(\beta_2)B_\psi(\sigma_e^2)\mathbf{B}d\beta_2d\beta_1 + 2\int_0^\infty \int_{\beta_1}^\infty \gamma(\beta_2)C_\psi(\sigma_e^2)\mathbf{B}\mathbf{v}\mathbf{v}^T\mathbf{B}d\beta_2d\beta_1, \tag{B-11}$$

with the boundary conditions:  $\bar{\mathbf{F}}(\infty) = 0$  and  $\partial\bar{\mathbf{F}}(\beta)/\partial\beta|_{\beta=\infty} = 0$ . For NLMS algorithm with general non-linearity, the integrals are rather difficult to evaluate because of the presence of  $\sigma_e^2$ .

Fortunately, it can be shown in Appendix C that for M-nonlinearities which include commonly used M-estimate functions and several practical nonlinearities,  $B_\psi(\sigma_e^2)/\sigma_e^2 = S_\psi(\sigma_e^2)$  and  $C_\psi(\sigma_e^2)$  are approximately independent of  $\sigma_e^2$  due to the use of ATS in (7)–(9) to update  $\xi$  and thus the estimated error variance  $\sigma_e^2$  is close to the “impulse-free” variance. Similar to the case of  $A_\psi(\hat{\sigma}_e^2)$ , if  $\xi$  is sufficiently large, then this is also a reasonable approximation, especially at the steady state. Thus,  $S_\psi(\sigma_e^2)$  and  $C_\psi(\sigma_e^2)$  can be taken outside of the integral if the above conditions are satisfied. This leaves the term  $\sigma_e^2$ , which is identical to the conventional analysis of the NLMS algorithm. Taking  $S_\psi(\sigma_e^2)$  and  $C_\psi(\sigma_e^2)$  outside the integral, one gets

$$s_3 \approx C_\psi(\sigma_e^2)\mathbf{I}_1 + S_\psi(\sigma_e^2)\mathbf{I}_2 + S_\psi(\sigma_e^2)\sigma_g^2\mathbf{I}_3. \tag{B-12}$$

where  $\mathbf{I}_1 = 2\int_0^\infty \int_{\beta_1}^\infty \gamma(\beta_2)\mathbf{B}\mathbf{v}\mathbf{v}^T\mathbf{B}d\beta_2d\beta_1$ ,  $\mathbf{I}_2 = \int_0^\infty \int_{\beta_1}^\infty \gamma(\beta_2)(\mathbf{v}^T\mathbf{B}\mathbf{v})\mathbf{B}d\beta_2d\beta_1$ , and  $\mathbf{I}_3 = \int_0^\infty \int_{\beta_1}^\infty \gamma(\beta_2)\mathbf{B}d\beta_2d\beta_1$ . The integrals  $\mathbf{I}_1 \sim \mathbf{I}_3$  also appear in the analysis of the NLMS algorithm in Gaussian noise [37] and they have been evaluated to be

$$\mathbf{I}_1 = \mathbf{U}\mathbf{D}_1\mathbf{U}^T, \tag{B-13}$$

where  $\mathbf{D}_1 = 2\Lambda[(\mathbf{V}\mathbf{V}^T) \circ \mathbf{I}(\Lambda)]\Lambda$ ,  $[\mathbf{I}(\Lambda)]_{ij} = I_{ij}(\Lambda)$ ,  $\circ$  denotes element-wise product of two matrices,

$$I_{ij}(\Lambda) = \int_0^\infty \beta \exp(-\beta\varepsilon) \left[ \prod_{k=1}^L (2\beta\lambda_k + 1)^{-1/2} \right] (2\beta\lambda_i + 1)^{-1} (2\beta\lambda_j + 1)^{-1} d\beta, \tag{B-14a}$$

and

$$\mathbf{I}_2 = \mathbf{U}\mathbf{D}_2\mathbf{U}^T \tag{B-14b}$$

where  $\mathbf{D}_2$  is a diagonal matrix and its  $i$ -th diagonal element is given by  $[\mathbf{D}_2]_{i,i} = \sum_k \lambda_k \lambda_i I_{ki}(\Lambda) [\mathbf{V}\mathbf{V}^T]_{k,k}$ .

$$\mathbf{I}_3 = \int_0^\infty \int_{\beta_1}^\infty \gamma(\beta_2)\mathbf{U}\mathbf{D}(n)\mathbf{U}^T d\beta_2d\beta_1 = \mathbf{U}\mathbf{D}_3(n)\mathbf{U}^T, \tag{B-15}$$

where  $\mathbf{D}_3(n) = \Lambda\mathbf{I}'(\Lambda)$  and  $\mathbf{I}'(\Lambda)$  is a diagonal matrix with its  $i$ -th diagonal element given by

$$I'_i(\Lambda) = \int_0^\infty \beta \exp(-\beta\varepsilon) \left[ \prod_{k=1}^L (2\beta\lambda_k + 1)^{-\frac{1}{2}} \right] (2\beta\lambda_i + 1)^{-1} d\beta. \tag{B-16}$$

Substituting (B-13)~(B-15) into (B-12) yields

$$\mathbf{M}_3 = \mu^2 E_{\{v\}}[s_3] \approx 2C_\psi(\sigma_e^2)\mu^2\mathbf{U}\{\Lambda[(\mathbf{U}^T\Xi(n)\mathbf{U}) \circ \mathbf{I}(\Lambda)]\}\mathbf{U}^T + S_\psi(\sigma_e^2)\mu^2\mathbf{U}\mathbf{D}_2\mathbf{U}^T + S_\psi(\sigma_e^2)\mu^2\sigma_g^2\mathbf{U}\Lambda\mathbf{I}'(\Lambda)\mathbf{U}^T. \tag{B-17}$$

This expression is somewhat complicated. To simplify the determination of the step size bound, we can alternatively adopt the argument in Appendix A and that in [37] by approximating  $\sigma_e^2$  inside the integral by the mean value of its upper bound  $\sigma_e^2(n) = E[\mathbf{v}^T\mathbf{R}_{XX}\mathbf{v}] + \sigma_g^2$ , which is the mean MSE at time  $n$ . Although this approximation is not entirely valid for all  $n$ ,  $B_\psi(\sigma_e^2)$  only dominates at the steady states where the other terms have died down. Therefore, the mean square behavior can still be satisfactorily described as shown in the simulation results. Moreover, in (B-17) since both terms are multiplied by  $\mu^2$  which is usually small, they only affect the behavior at the steady state where the approximation is more likely to be valid. By doing so, the last two terms in (B-17) will be tightly bounded by  $\mu^2 S_\psi(\sigma_e^2)\sigma_e^2(n)\mathbf{U}\Lambda\mathbf{I}'(\Lambda)\mathbf{U}^T$ .

Finally, we note that for the LMS algorithm, the normalization term in (B-2) is missing and  $\mathbf{L}_3$  is obtained by replacing  $\mathbf{B}$  above by  $\mathbf{R}_{XX}$ .

### Appendix C: M-Estimate Functions (Nonlinearities) and Properties

In addition to the modified Huber M-estimate function, the bi-weight family of functions and the Hampel’s three parts redescending function are also commonly used in robust statistics [40]. The score functions of these M-estimate functions can be written as

$$\rho'_\xi(e) = \psi_\xi(e) = eq(e/\xi), \tag{C-1}$$

where  $q(e/\xi)$  is the score function and  $\xi$  is a threshold for controlling the tradeoff between degree of outlier suppression. We call those nonlinearities satisfying (C-1) the M-nonlinearity. As an M-estimate score function,  $\psi_\xi(e)$  should be equal to  $e$  when  $|e|$  is much smaller than  $\xi$  and gradually

decrease in magnitude as  $|e|$  increases, i.e.  $q(0) = 1$ . For redescending score function,  $\psi(\infty) = 0$  so that outliers with very large amplitude are completely ignored. The score functions for the bi-weight family of functions, the Hampel’s function and the MH nonlinearity are all redescending. More specifically:

*Bi-Weight Family of Functions*

$$\rho_{\text{BW}}(e) = \begin{cases} (\xi^2/6) \left\{ 1 - \left[ 1 - (e/\xi)^2 \right]^3 \right\}, & \text{if } |e| \leq \xi \\ 1, & \text{if } |e| > \xi \end{cases} \quad (\text{C-2})$$

$$\rho'_{\text{BW}}(e) = \psi_{\text{BW}}(e) = e \left[ 1 - (e/\xi)^2 \right]^2, I(|e| \leq \xi) \text{ and } 0 \text{ elsewhere where } I(\cdot) \text{ is the indicator function.}$$

*Hampel’s Function*

$$\psi_{\text{HP}}(e) = e q_{\text{HP}}(e/\xi), \quad (\text{C-3})$$

$$q_{\text{HP}}(e) = \begin{cases} 1, & 0 \leq |e| < 1 \\ \text{sgn}(e)/e, & 1/1 \leq |e| < \xi_1/\xi \\ \text{sgn}(e)(\text{sgn}(e) - \xi_2)/e(\xi_1 - \xi_2), & \xi_1/\xi \leq |e| < \xi_2/\xi \\ 0, & \xi_2/\xi \leq |e| \end{cases}$$

The score function of the EF nonlinearity and the Huber function also satisfy (C-1), but they are not redescending. Therefore, they will be sensitive to outliers with large amplitude. More specifically, the weighting functions of the EF nonlinearity is:

$$\psi_{\text{EF}}(e) = e q_{\text{EF}}(e/\xi),$$

where  $q_{\text{EF}}(e) = \frac{1}{e} \int_0^e \exp(-u^2/2) du$ . Note, for the sake of presentation we have used  $\xi$  instead of  $\sigma_y$ , as shown in Table 1. This is also true for the quantizer function in which

$$\psi_{\text{Q}}(\mu e) = (\mu e) q_{\text{Q}}(\mu e / \Delta),$$

$$\text{where } q_{\text{Q}}(\mu e) = \begin{cases} 0, & |\mu e| < 1/2 \\ (1/\mu e) \text{sgn}(\mu e), & 1/2 \leq |\mu e| < 1 \\ 1, & \text{otherwise} \end{cases}$$

We now show that  $A_{\psi}(\sigma_e^2)$ ,  $B_{\psi}(\sigma_e^2)/\sigma_e^2$ , and  $C_{\psi}(\sigma_e^2)$  are approximately independent of  $\sigma_e^2$ , if the score function has the form in (C-1) and the ATS is employed. First of all, we note from (C-1) that

$$\psi'_{\xi}(e) = q(e/\xi) + (e/\xi)q'(e/\xi). \quad (\text{C-4})$$

Using change of variable  $u = e/\sigma_e$ , one gets

$$E[\psi'_{\xi}(e)] = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} \left[ q\left(\frac{u\sigma_e}{\xi}\right) + \left(\frac{e}{\xi}\right)q'\left(\frac{u\sigma_e}{\xi}\right) \right] \exp\left(-\frac{u^2}{2}\right) du. \quad (\text{C-5})$$

If ATS is employed, then  $\xi = k_{\xi} \hat{\sigma}_e$ . Assuming that  $\hat{\sigma}_e \approx \kappa \sigma_e$ , i.e. the estimate “impulse free” noise standard deviation is  $\kappa$  times of  $\sigma_e$ , then (C-5) becomes

$$E[\psi'_{\xi}(e)] = A_{\psi}(\sigma_e^2) \approx \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} \left( q\left(\frac{u}{\kappa k_{\xi}}\right) + \left(\frac{e}{\xi}\right)q'\left(\frac{u}{\kappa k_{\xi}}\right) \right) \exp\left(-\frac{u^2}{2}\right) du$$

which is almost independent of  $\sigma_e^2$ . For normal operation without outliers,  $\kappa \approx 1$ . In the presence of impulses, the conditioned error variance  $\sigma_e$  will be much larger than the “impulse free” error variance and  $\kappa \gg 1$ . Consequently,  $A_{\psi}(\sigma_e^2)$  will be nearly zero. Similarly, we have

$$B_{\psi}(\sigma_e^2) = \frac{1}{\sqrt{2\pi}\sigma_e} \int_{-\infty}^{\infty} e^2 q^2(e/\xi) \exp\left(-\frac{e^2}{2\sigma_e^2}\right) du \sigma_e \\ = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} u^2 \sigma_e^2 q^2(u\sigma_e/\xi) \exp\left(-\frac{u^2}{2}\right) du.$$

Using ATS and assuming  $\hat{\sigma}_e \approx \kappa \sigma_e$  yields  $B_{\psi}(\sigma_e^2) = \sigma_e^2 S_{\psi}(\sigma_e^2) \approx \sigma_e^2 S_{\psi}(\kappa)$ , where  $S_{\psi}(\kappa) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} u^2 q^2\left(\frac{u}{\kappa k_{\xi}}\right) \exp\left(-\frac{u^2}{2}\right) du$  is independent of  $\sigma_e$ . Note,  $B_{\psi}(\sigma_e^2)$  is proportional to the error power  $\sigma_e^2$ . Finally,

$$C_{\psi}(\sigma_e^2) \approx \frac{d}{d\sigma_e^2} E[\psi^2(e)] = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} u^2 q^2\left(\frac{u\sigma_e}{\xi}\right) \exp\left(-\frac{u^2}{2}\right) du \\ + \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} \left(\frac{u^3 \sigma_e}{\xi}\right) q\left(\frac{u\sigma_e}{\xi}\right) q'\left(\frac{u\sigma_e}{\xi}\right) \exp\left(-\frac{u^2}{2}\right) du.$$

If ATS is employed and assuming that  $\hat{\sigma}_e \approx \kappa \sigma_e$ , then

$$C_{\psi}(\sigma_e^2) \approx S_{\psi}(\kappa) + \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} \left(\frac{u^3}{\kappa k_{\xi}}\right) q\left(\frac{u}{\kappa k_{\xi}}\right) q'\left(\frac{u}{\kappa k_{\xi}}\right) \exp\left(-\frac{u^2}{2}\right) du,$$

which is again independent of  $\sigma_e$ . In summary, if ATS is employed,  $A_{\psi}(\sigma_e^2)$ ,  $C_{\psi}(\sigma_e^2)$ , and  $S_{\psi}(\sigma_e^2)$  depend very weakly on  $\sigma_e^2$  and they depend on the threshold parameters  $k$  and the ratio between  $\sigma_e^2$  and the “impulse free” noise variance  $\hat{\sigma}_e^2$ ,  $\kappa^2$ . For normal operation without outliers,  $\kappa \approx 1$ . In case of impulsive outliers,  $\kappa^2$  will be much larger than one, and their values will be very small.

**References**

1. Widrow, B., McCool, J., Larimore, M. G., & Johnson, C. R., Jr. (1976). Stationary and nonstationary learning characteristics of the LMS adaptive filter. *Proceedings of the IEEE*, 64, 1151–1162.
2. Plackett, R. L. (1972). The discovery of the method of least-squares. *Biometrika*, 59(2), 239–251.
3. Nagumo, J. I., & Noda, A. (1967). A learning method for system identification. *IEEE Transactions on Automatic Control*, AC-12, 282–287.
4. Bitmead, R. R., & Anderson, B. D. O. (1980). Performance of adaptive estimation algorithms in dependent random environments. *IEEE Transactions on Automatic Control*, 25, 788–794.
5. Haykin, S. (2001). *Adaptive filter theory* (4th ed.). Englewood Cliffs: Prentice-Hall.

6. Zou, Y. (2000). Robust statistical based adaptive filtering algorithms for impulsive noise suppression. Hong Kong: Ph.D. Dissertation, The Univ. Hong Kong.
7. Zou, Y., Chan, S. C., & Ng, T. S. (2000). Least mean M-estimate algorithms for robust adaptive filtering in impulsive noise. *IEEE Transactions on Circuits and Systems II*, 47(12), 1564–1569.
8. Haweel, T. I., & Clarkson, P. M. (1992). A class of order statistic LMS algorithms. *IEEE Transactions on Signal Processing*, 40(1), 44–53.
9. Settineri, R., Najim, M., & Ottaviani, D. (1996). Order statistic fast Kalman filter. In *Proceedings of IEEE International Symposium on Circuits and Systems* (Vol. 2, pp. 116–119).
10. Koike, S. (1997). Adaptive threshold nonlinear algorithm for adaptive filters with robustness against impulsive noise. *IEEE Transactions on Signal Processing*, 45(9), 2391–2395.
11. Weng, J. F., & Leung, S. H. (1997). Adaptive nonlinear RLS algorithm for robust filtering in impulsive noise. In *Proceedings of IEEE International Symposium on Circuits and Systems* (Vol. 4, pp. 2337–2340).
12. Chan, S. C., & Zou, Y. (2004). A recursive least M-estimate algorithm for robust adaptive filtering in impulsive noise: fast algorithm and convergence performance analysis. *IEEE Transactions on Signal Processing*, 52(4), 975–991.
13. Huber, P. J. (1981). *Robust statistics*. New York: Wiley.
14. Horowitz, L. L., & Senne, K. D. (1981). Performance advantage of complex LMS for controlling narrowband adaptive arrays. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 29(3), 722–736.
15. Feuer, A., & Weinstein, E. (1985). Convergence analysis of LMS filters with uncorrelated Gaussian data. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 33(1), 222–230.
16. Foley, J. B., & Boland, F. M. (1988). A note on the convergence analysis of LMS adaptive filters with Gaussian data. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 36(7), 1087–1089.
17. Bershad, N. J. (1986). Analysis of the normalized LMS algorithm with Gaussian inputs. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, ASSP-34(4), 793–806.
18. Bershad, N. J. (1987). Behavior of the  $\epsilon$ -normalized LMS algorithm with Gaussian inputs. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, ASSP-35(5), 636–644.
19. Tarrab, M., & Feuer, A. (1988). Convergence and performance analysis of the normalized LMS algorithm with uncorrelated Gaussian data. *IEEE Transactions on Information Theory*, IT-34(4), 680–691.
20. Rupp, M. (1993). The behavior of LMS and NLMS algorithms in the presence of spherically invariant processes. *IEEE Transactions on Signal Processing*, 41(3), 1149–1160.
21. Slock, D. T. M. (1993). On the convergence behavior of the LMS and the normalized LMS algorithms. *IEEE Transactions on Signal Processing*, 41(9), 2811–2825.
22. Costal, M. H., & Bermudez, J. C. M. (2002). An improved model for the normalized LMS algorithm with Gaussian inputs and large number of coefficients. In *Proceedings of IEEE ICASSP'2002* (Vol. 2, pp. 1385–1388).
23. Barrault, G., Costa, M. H., J. Bermudez, C. M. & Lenzi, A. (2005). A new analytical model for the NLMS algorithm. In *Proc. IEEE Int. Conf. Acoust., Speech, Signal Processing* (Vol. 4, pp. 41–44).
24. Lobato, E. M., Tobias, O. J., & Seara, R. (2006). Stochastic model for the NLMS algorithm with correlated Gaussian data. In *Proc. IEEE Int. Conf. Acoust., Speech, Signal Processing* (Vol. 3, pp. 760–763).
25. Sayed, A. H., & Rupp, M. (1995). A time-domain feedback analysis of adaptive gradient algorithms via the small gain theorem. In *Proc. SPIE Conf. Advanced Signal Processing: Algorithms, Architectures, Implementations* (Vol. 2563, pp. 458–469). San Diego.
26. Yousef, N. R., & Sayed, A. H. (2001). A unified approach to the steady-state and tracking analyses of adaptive filters. *IEEE Transactions on Signal Processing*, 49(2), 314–324.
27. Al-Naffouri, T. Y., & Sayed, A. H. (2003). Transient analysis of data-normalized adaptive filters. *IEEE Transactions on Signal Processing*, 51(3), 639–663.
28. Koike, S. (2006). Convergence analysis of adaptive filters using normalized sign-sign algorithm. *IEICE Transactions on Fundamentals of Electronics, Communications and Computer Sciences*, E88-A(11), 3218–3224.
29. Koike, S. (2006). Performance analysis of the normalized LMS algorithm for complex-domain adaptive filters in the presence of impulse noise at filter input. *IEICE Transactions on Fundamentals of Electronics, Communications and Computer Sciences*, E89-A(9), 2422–2428.
30. Tukey, J. W. (1960). A survey of sampling from contaminated distributions in contributions to probability and statistics. I. Olkin (Ed.), Stanford University Press.
31. Price, R. (1958). A useful theorem for nonlinear devices having Gaussian inputs. *IRE Transactions on Information Theory*, IT-4, 69–72.
32. McMahan, E. L. (1964). An extension of Price's theorem. *IEEE Transactions on Information Theory*, IT-10, 168.
33. Papoulis, A. (1965). Comment on 'an extension of Price's theorem'. *IEEE Transactions on Information Theory*, IT-11, 154.
34. Mathews, V. J. (1991). Performance analysis of adaptive filters equipped with the dual sign algorithm. *IEEE Transactions on Signal Processing*, 39(1), 85–91.
35. Bershad, N. J. (1988). On Error-saturation nonlinearities in LMS adaptation. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, ASSP-36(4), 440–452.
36. Price, R. (1964). Comment on: 'a useful theorem for nonlinear devices having Gaussian inputs'. *IEEE Transactions on Information Theory*, IT-10, 171.
37. Chan, S. C., & Zhou, Y. (2009). Convergence behavior of NLMS algorithm for Gaussian inputs: Solutions using generalized Abelian integral functions and step size selection. To appear in *Journal of Signal Processing Systems*, 2010.
38. Siegel, C. L. (1988). *Topics in complex function theory, vol. 2: Automorphic functions and Abelian Integrals*. New York: Wiley.
39. Macleod, M. D. (2005). Robust normalized LMS filtering. In *Proc. IEEE Int. Conf. Acoust., Speech, Signal Processing* (Vol. 4, pp. 37–40).
40. Hampel, F. R., Ronchetti, E. M., Rousseeuw, P. J., & Stahel, W. A. (2005). *Robust statistics: The approach based on influence functions*. New York: Wiley.
41. Zhou, Y. (2006). Improved analysis and design of efficient adaptive transversal filtering algorithms with particular emphasis on noise, input and channel modeling. Ph.D. Dissertation, The Univ. Hong Kong, Hong Kong.
42. Douglas, S. C., & Meng, T. H. Y. (1994). Normalized data nonlinearities for LMS adaptation. *IEEE Transactions on Signal Processing*, 42(6), 1352–1365.
43. Gersho, A. (1984). Adaptive filtering with binary reinforcement. *IEEE Transactions on Information Theory*, IT-30(2), 191–199.
44. Doob, J. L. (1948). Asymptotic properties of Markoff transition probabilities. *Transactions of the American Mathematical Society*, 63, 393–421.
45. Recktenwald, G. (2000). *Numerical methods with MATLAB: Implementations and applications*. Englewood Cliffs: Prentice-Hall.
46. Chan, S. C., & Zhou, Y. (2007). On the convergence analysis of the normalized LMS and the normalized least mean M-estimate algorithms. In *Proc. IEEE International Symposium on Signal Processing and Information Technology* (pp. 1059–1065).
47. Jageman, O. L. (1975). Nonstationary blocking in telephone traffic. *Bell System Technical Journal*, 54(3), 625–661.



**S. C. Chan** (S'87–M'92) received the B.Sc. (Eng.) and Ph.D. degrees from the University of Hong Kong in 1986 and 1992, respectively. He joined City Polytechnic of Hong Kong in 1990 as an Assistant Lecturer and later as a University Lecturer. Since 1994, he has been with the Department of Electrical and Electronic Engineering, the University of Hong Kong, and is now a Professor. He held visiting positions in Microsoft Corporation, Redmond, USA, Microsoft Research Asia, University of Texas at Arlington, and Nanyang Technological University. Dr. Chan's main research area is signal processing and applications. His previous research covers fast transform algorithms, filter design and realization, multirate signal processing, adaptive and communications signal processing, data compression and image-based rendering. Dr. Chan has served in a number of professional and administrative committees. He is currently a member of the Digital Signal Processing Technical Committee of the IEEE Circuits and Systems Society, Associate Editors for IEEE Trans. Circuits and Systems I and Journal of Signal Processing systems, and was the Chairman of the IEEE HK Chapter of Signal Processing. He

was the special session chairman of ICASSP2003 and is an organizing committee member of the IEEE 2010 ICIP.



**Y. Zhou** received the B.Sc., M.Sc. degrees in 1997 and 1999 respectively, both in control engineering from Harbin Institute of Technology, Harbin, China, and Ph.D. degree in electrical engineering in 2006, from the University of Hong Kong, Hong Kong, China. From July 2006 to May 2009, he was with the Department of Electrical and Electronic Engineering, the University of Hong Kong as a postdoctoral research fellow. His research interests are adaptive signal processing, system identification, statistical signal processing and estimation theory. Dr. Zhou joined the Key Laboratory of Noise and Vibration Research, Institute of Acoustics, Chinese Academy of Sciences in August 2009, where he is now an associate professor and engaged in research on statistical signal processing theory and techniques applied to acoustics. E-mail: yizhou@mail.ioa.ac.cn