A Recursive Least M-Estimate Algorithm for Robust Adaptive Filtering in Impulsive Noise: Fast Algorithm and Convergence Performance Analysis

Shing-Chow Chan, *Member, IEEE,* and Yue-Xian Zou

Abstract—This paper studies the problem of robust adaptive filtering in impulsive noise environment using a recursive least M-estimate algorithm (RLM). The RLM algorithm minimizes a robust M-estimator-based cost function instead of the conventional mean square error function (MSE). Previous work has showed that the RLM algorithm offers improved robustness to impulses over conventional recursive least squares (RLS) algorithm. In this paper, the mean and mean square convergence behaviors of the RLM algorithm under the contaminated Gaussian impulsive noise model is analyzed. A lattice structure-based fast RLM algorithm, called the Huber Prior Error Feedback-Least Squares Lattice (H-PEF-LSL) algorithm[1] is derived. It has an order $O(N)$ arithmetic complexity, where N is the length of the adaptive filter, and can be viewed as a fast implementation of the RLM algorithm based on the modified Huber M-estimate function and the conventional PEF-LSL adaptive filtering algorithm. Simulation results show that the transversal RLM and the H-PEF-LSL algorithms have better performance than the conventional RLS and other RLS-like robust adaptive algorithms tested when the desired and input signals are corrupted by impulsive noise. Furthermore, the theoretical and simulation results on the convergence behaviors agree very well with each other.

Index Terms—Adaptive filter, contaminated Gaussian distribution, impulsive noise suppression, lattice structure, Prior Error Feedback-Least Square Lattice Algorithm, recursive least M-estimate algorithm, robust statistics, system identification.

I. INTRODUCTION

CONVENTIONAL adaptive filters minimizing least squares (LS) or mean square criterion are very sensitive to impulsive noise. This is of increasing importance in modern communication systems, where the performance is usually limited by interference of impulsive nature. Such impulsive noise, which results from natural or man-made electromagnetic waves [1], usually has a long tail distribution and violates the commonly used Gaussian noise assumption. Under such circumstances, the performance of linear adaptive filters will deteriorate significantly. Nonlinear techniques are often

employed to reduce the hostile effect on the system due to such impulsive interference. For example, in the order-statistic least mean square LMS (OSLMS) and the order-statistic fast Kalman filtering (OSFKF) algorithms [2], [3], median filtering is applied to the conventional LMS and the recursive least squares (RLS) algorithms to protect the filter weights from the adverse effects of impulsive noise. Similarly, in the adaptive threshold nonlinear (ATNA) and nonlinear RLS (N-RLS) algorithms [4], [5], nonlinear clipping functions are used to limit the transient fluctuation of the estimation error in conventional adaptive filters caused by the impulses. Using a cost function combining the L_1 and the L_2 norms, a mixed-norm LMS (RMN) algorithm has also been proposed for combating impulsive noise [6]. Recently, the authors have proposed a RLS-like algorithm, called the recursive least M-estimate (RLM) algorithm, for impulsive noise suppression [7]. It minimizes a new cost function based on robust M-estimate functions, instead of the conventional mean square error (MSE). Simulation results showed that the RLM algorithm is more robust than the conventional RLS, N-RLS, RMN, OSFKF, and ATNA algorithms when the input and desired signals are corrupted by contaminated Gaussian (CG) noise or alpha-stable distributed noise [7], [11], [16]. Like the RLS algorithm, the RLM algorithm has a computational complexity of order $O(N^2)$, where N is the number of taps in the adaptive transversal filter.

In this paper, we analytically analyze the convergence of the RLM algorithm under the contaminated Gaussian impulsive noise model and show, both theoretically and experimentally, its robustness over the conventional RLS algorithm. Both mean and mean square convergence are studied. A lattice structure based fast RLM algorithm, called the Huber Prior Error Feedback-Least Squares Lattice (H-PEF-LSL) algorithm, with order $O(N)$ arithmetic complexity, is also derived, which can be viewed as a fast implementation of the RLM algorithm based on the modified Huber M-estimate function and the conventional PEF-LSL adaptive filtering algorithm. Extensive computer simulations are also carried out to evaluate the performance of the proposed RLM and H-PEF-LSL algorithms under impulsive noise environment. Simulation results show that the transversal RLM and the H-PEF-LSL algorithms have better performance than the conventional RLS and other RLS-like robust adaptive algorithms when the desired and input signals are corrupted by impulsive noise. Moreover, it is found that the theoretical and simulation results on the mean and mean square convergence of the RLM algorithm agree very well with each other.

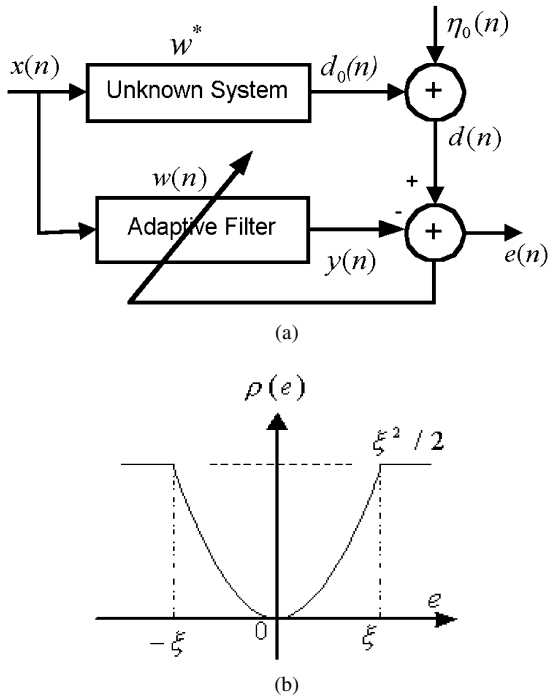[1]Part of the H-PEF-LSL algorithm was presented in ICASSP'2001.

(a)

(b)

Fig. 1(a). System identification structure. (b) Modified Huber M-estimate function.

The paper is organized as follows. In Section II, the principle of the robust M-estimate filtering and the RLM algorithm are briefly reviewed. The proposed Huber PEF-LSL algorithm is then derived in Section III. Section IV presents the mean and mean square convergence analyses of the proposed RLM algorithm. The simulation results and performance comparison of the proposed RLM and H-PEF-LSL algorithms with other conventional algorithms are evaluated in Section V. Finally, conclusions are drawn in Section VI.

II. TRANSVERSAL RECURSIVE LEAST M-ESTIMATE ALGORITHM

Without loss of the generality, let us consider the system identification problem shown in Fig. 1(a). The signals $x(n)$ and $y(n)$ are, respectively, the input and output of the linear transversal adaptive filter. The estimation error is given by $e(n) = d(n) - \boldsymbol{w}^T(n)\boldsymbol{X}(n)$, where the superscript T denotes the transposition operation. N, $\boldsymbol{w}(n) = [w_1(n), \cdots, w_N(n)]^T$, and $\boldsymbol{X}(n) = [x(n), \cdots, x(n-N+1)]^T$ are, respectively, the filter length, the filter weight and the input vectors. The observed desired signal $d(n) = d_0(n) + \eta_o(n)$ is assumed to be the sum of the output of the unknown system $d_0(n)$ and the interference noise $\eta_o(n)$. To provide robust filtering in impulsive noise environment, the following cost function, instead of the conventional LS cost function $J_{LS}(n) = \sum_{i=1}^{n} \lambda^{n-i} e^2(i)$, was proposed by the authors in [7]:

$$J_\rho(n) \triangleq \sum_{i=1}^{n} \lambda^{n-i} \rho\left(e(i)\right) = \sum_{i=1}^{n} \lambda^{n-i} \rho\left(d(i) - \boldsymbol{w}^T(n)\boldsymbol{X}(i)\right)$$

(1)

where λ is the positive forgetting factor, and $\rho(\cdot)$ is an M-estimate function. The purpose of using an M-estimate function $\rho(\cdot)$, instead of the squaring function in the conventional LS cost function, is to limit the adverse effect of impulsive noise on the cost function, when the error signal becomes very large. Fig. 1(b) shows the following modified Huber M-estimate function used in this paper.

$$\rho(e) = \begin{cases} \frac{e^2}{2}, & 0 < |e| < \xi \\ \frac{\xi^2}{2}, & \text{otherwise} \end{cases}$$

(2)

where ξ is the threshold parameter. It can be seen that the function $\rho(\cdot)$ is a real-valued even function and is quadratic when e is smaller than ξ. However, for values greater than ξ, $\rho(\cdot)$ is equal to a constant, which helps to suppress impulses with large amplitude. The degree of impulsive noise suppression, on the other hand, is controlled by the threshold parameter ξ. The smaller the value of ξ, the greater will be the suppression of the impulses. The threshold parameter is usually chosen according to the application at hand or estimated continuously. This will be discussed later in this section. From the above discussion, it is apparent that the objective function $J_\rho(n)$, as defined in (1), can be used to smooth out momentary fluctuation due to impulsive noise. In [7], $\rho(\cdot)$ is chosen as a class of M-estimates called the Hampel's three part redescending function. The M-estimate, or maximum likelihood-type estimator, was originally proposed to improve the robustness of statistical estimators subject to small deviations from the assumption in [10, pp. 1]. An important example is distributional robustness, which is concerned with slight deviation from the assumed model (usually the Gaussian distribution). If $\rho(\boldsymbol{x}, \theta)$ is chosen as $-\log f(\boldsymbol{x}, \theta)$, the M-estimate gives the ordinary maximum likelihood estimate (MLE) [10, pp. 43], where $\boldsymbol{x}$ is the observed random variable with probability density function (pdf) $f(\boldsymbol{x})$, and θ is the parameter vector to be estimated. In practical situations, the underlying pdf of the noises are difficult to estimate, and $\rho(\boldsymbol{x}, \theta)$ is usually chosen as a fixed function of $\boldsymbol{x}$ only, that is, $\rho(\boldsymbol{x})$. Hence, the M-estimate approximates the MLE, where the pdfs are assumed to be known exactly. More general redescending M-estimators can be applied, and they have similar behaviors [9], [10]. The main reason for choosing the modified Huber M-estimate function in this paper is due to its good performance and simplicity, which makes the order and time updates in the conventional PEF-LSL algorithm feasible. The performance analysis is also simplified considerably. This will be further elaborated upon in Section III. Simulation results showed that the performance of the RLM algorithm using the modified Huber M-estimate function and the Hampel's three part redescending function are very close to each other under the contaminated Gaussian and the alpha-stable distributed noise environment [11].

The optimal solution $\boldsymbol{w}(n)$ for minimizing $J_\rho(n)$ can be obtained by differentiating (1) with respect to $\boldsymbol{w}(n)$ and setting the derivatives to zero. This yields the following M-estimate normal equation:

$$\boldsymbol{R}_{\boldsymbol{X}\rho}(n)\boldsymbol{w}(n) = \boldsymbol{P}_{\boldsymbol{X}\rho}(n)$$

(3)

where

$$\boldsymbol{R}_{\boldsymbol{X}\rho}(n) = \sum_{i=1}^{n} \lambda^{n-i} q\left(e(i)\right) \boldsymbol{X}(i) \boldsymbol{X}^T(i)$$

$$= \lambda \boldsymbol{R}_{\boldsymbol{X}\rho}(n-1) + q\left(e(n)\right) \boldsymbol{X}(n) \boldsymbol{X}^T(n) \quad (4)$$

$$\boldsymbol{P}_{\boldsymbol{X}\rho}(n) = \sum_{i=1}^{n} \lambda^{n-i} q\left(e(i)\right) d(i) \boldsymbol{X}(i)$$

$$= \lambda \boldsymbol{P}_{\boldsymbol{X}\rho}(n-1) + q\left(e(n)\right) d(n) \boldsymbol{X}(n). \quad (5)$$

Moreover, $q(e) \triangleq \psi(e)/e$ and $\psi(e) \triangleq d\rho(e)/de$ are the weighting function and score function of $\rho(\cdot)$, respectively, whereas $\boldsymbol{R}_{\boldsymbol{X}\rho}(n)$ and $\boldsymbol{P}_{\boldsymbol{X}\rho}(n)$ are called the M-estimate correlation matrix of $\boldsymbol{X}(n)$ and the M-estimate cross-correlation vector of $\boldsymbol{X}(n)$ and $d(n)$, respectively. They serve similar purpose as the conventional correlation matrix of $\boldsymbol{X}(n)$ and the cross-correlation vector of $\boldsymbol{X}(n)$ and $d(n)$, respectively. Applying the matrix inversion lemma [8, pp. 565] $(\boldsymbol{A} + \mu \boldsymbol{x}\boldsymbol{y}^T)^{-1} = \boldsymbol{A}^{-1}(\boldsymbol{I} - (\mu \boldsymbol{x}\boldsymbol{y}^T \boldsymbol{A}^{-1}/1 + \mu \boldsymbol{y}^T \boldsymbol{A}^{-1}\boldsymbol{x}))$ to (4) and letting $\boldsymbol{R}_{\boldsymbol{X}\rho}(n-1) = \lambda \boldsymbol{A}$, $\boldsymbol{x} = \boldsymbol{y} = \boldsymbol{X}(n)$, and $\mu = q(e(n))$, the following recursive Least M-estimate (RLM) algorithm is derived for solving the M-estimate normal equation in (3):

$$\boldsymbol{V}(n) = \lambda^{-1} \left(\boldsymbol{I} - \boldsymbol{K}(n)\boldsymbol{X}^T(n)\right) \boldsymbol{V}(n-1) \quad (6)$$

$$\boldsymbol{K}(n) = \frac{q\left(e(n)\right)\boldsymbol{V}(n-1)\boldsymbol{X}(n)}{\lambda + q\left(e(n)\right)\boldsymbol{X}^T(n)\boldsymbol{V}(n-1)\boldsymbol{X}(n)} \quad (7)$$

$$\boldsymbol{w}(n) = \boldsymbol{w}(n-1) + \left(d(n) - \boldsymbol{X}^T(n)\boldsymbol{w}(n-1)\right) \boldsymbol{K}(n) \quad (8)$$

where $\boldsymbol{V}(n) = \boldsymbol{R}_{\boldsymbol{X}\rho}^{-1}(n)$, and $\boldsymbol{K}(n)$ is the M-estimate gain vector. It can be observed that when $e(n)$ is smaller than ξ, the weight function $q(e(n))$ is equal to one, and (7) becomes identical to the gain vector in the conventional RLS algorithm. However, when $e(n)$ is larger than ξ, $q(e(n))$, and hence, $\boldsymbol{K}(n)$ are equal to zero. From (6), it can be seen that the impulse corrupted input vector is prevented from entering $\bar{\boldsymbol{V}}(n)$, if $e(n)$ is larger than the threshold ξ. At the same time, the filter weight is not updated to prevent it from being affected by the possibly impulse corrupted desired or input signal vector. These properties make the proposed RLM algorithm more robust to the N-RLS algorithm, which only limits, but does not eliminate, the effect of the impulses. As mentioned earlier, the choice of the threshold parameter ξ can significantly affect the performance of the RLM algorithm. In [7], a systematic approach for estimating the threshold ξ was proposed. More precisely, the error signal is assumed, for simplicity, to be a mixture of Gaussian distributed random process (the so-called "impulse-free" signal or component) and additive impulsive noise. Suppose we can estimate the variance of this "impulse-free" component $\hat{e}(n)$. Then, the probability $P_r\{|\hat{e}(n)| > \xi\} = \theta_\xi = erfc\{\xi/(\sqrt{2}\hat{\sigma}(n))\}$, where $erfc(r) = 2\int_0^r e^{-x^2}dx/\sqrt{\pi}$ is the complementary error function [15], because of the Gaussian assumption of $\hat{e}(n)$. By choosing different values of θ_ξ, we have different confidence in detecting the impulsive noise that appears either at the desired or the input signals. For example, if θ_ξ is chosen to be 0.01, we have 99% confidence to detect and

reject the impulse. In this case, the threshold ξ in (2) can be chosen as

$$\xi = k_\xi \hat{\sigma}(n) = 2.576 \cdot \hat{\sigma}(n) \quad (9)$$

where $k_\xi = 2.576$, and $\hat{\sigma}^2(n)$ is the estimated variance of the "impulse-free" estimation error. Other values of θ_ξ and k_ξ can be used according to the degree of impulsive noise suppression. From experimental results, it is observed that the performance of RLM algorithm is not sensitive to the selection of k_ξ, provided they are not at the tail part of the distribution. To estimate $\hat{\sigma}^2(n)$, the following formula from [7] can be used:

$$\hat{\sigma}^2(n) = \lambda_\sigma \hat{\sigma}^2(n-1) + (1 - \lambda_\sigma)c_1 med\left(A_e(n)\right) \quad (10)$$

where $A_e(n) = \{e^2(n), \ldots, e^2(n - N_w + 1)\}$, med$(\cdot)$ denotes the sample median operation [12], N_w is the length of the estimation window, λ_σ is the forgetting factor, and $c_1 = 1.483(1 + 5/(N_w - 1))$ is a finite sample correction factor [12, pp. 44]. The term $c_1 med(A_e(n))$ is the robust instantaneous estimate of $\hat{\sigma}^2(n)$, where the medium operation helps to remove the additive impulsive noise. Due to the recursive nature of the estimation in (10), the estimation window is of infinite length, giving rise to a more stable estimation against impulsive noise. Computer simulations also show that the tracking ability of this estimation is very good. See [7], [11] for more details.

Comparing the RLM and the RLS algorithms, it can be seen that the computational complexity of the RLM algorithm is similar to the conventional RLS algorithm, requiring order $O(N^2)$ arithmetic operations and N more multiplications in (7) for computing $\boldsymbol{K}(n)$ and $O(N_w \log N_w)$ operations in (10) for computing $\hat{\sigma}^2(n)$. In the following section, a lattice structure-based fast RLM algorithm called the Huber Prior Error Feedback-Least Square Lattice (H-PEF-LSL) algorithm with $O(N)$ complexity is proposed.

III. HUBER PRIOR ERROR FEEDBACK-LEAST SQUARE LATTICE (H-PEF-LSL) ALGORITHM

In conventional adaptive filtering using the LS cost function, $\rho(\cdot)$ in (1) is chosen as the squaring function, that is, $J_\rho(n) = J_{LS}(n) = \sum_{i=1}^{n} \lambda^{n-i} e^2(i)$. The corresponding optimal solution $\hat{\boldsymbol{w}}(n)$ is governed by the following normal equation [8]:

$$\boldsymbol{R}(n)\hat{\boldsymbol{w}}(n) = \boldsymbol{P}(n) \text{ or } \hat{\boldsymbol{w}}(n) = \boldsymbol{R}^{-1}(n)\boldsymbol{P}(n) \quad (11)$$

where

$$\boldsymbol{R}(n) = \sum_{i=1}^{n} \lambda^{n-i} \boldsymbol{X}(i)\boldsymbol{X}^T(i) = \lambda \boldsymbol{R}(n-1) + \boldsymbol{X}(n)\boldsymbol{X}^T(n) \quad (12)$$

and

$$\boldsymbol{P}(n) = \sum_{i=1}^{n} \lambda^{n-i} d(i)\boldsymbol{X}(i) = \lambda \boldsymbol{P}(n-1) + d(n)\boldsymbol{X}(n) \quad (13)$$

are the autocorrelation matrix of $\boldsymbol{X}(n)$, and the cross-correlation vector between $\boldsymbol{X}(n)$ and $d(n)$, respectively. In the transversal RLS algorithm, the time recursive property of $\boldsymbol{R}(n)$ in (12) is used to reduce the arithmetic complexity to $O(N^2)$. The lattice structure-based LSL-type algorithms explore both the time-recursive and order-recursive properties of $\boldsymbol{R}(n)$, and
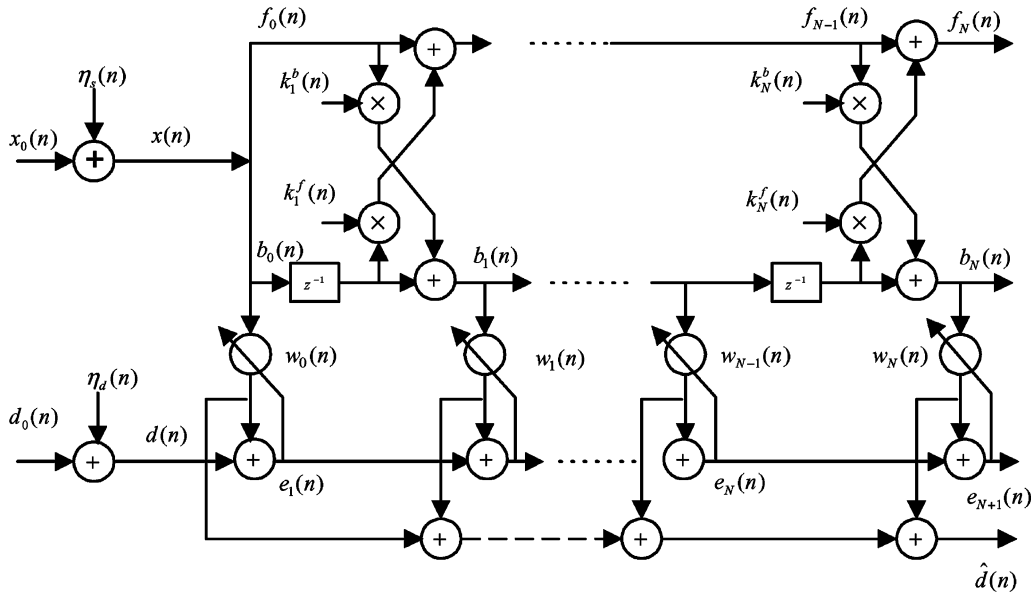
$$E_{0:N-1}^{f}(0) = E_{0:N-1}^{b}(-1) = \delta \ , \ \gamma_0(0) = 1 \ , \ w_{0:N}(0) = 0 \ , \ k_{1:N}^{f}(0) = k_{1:N}^{b}(0) = 0 \ ; \ \left(n > 0 \ , \ m = 1, \cdots N \right):$$

$$\gamma_0(n-1) = 1 \ , \ e_0(n) = d(n) \ , \ f_0(n) = b_0(n) = x(n) \ ,$$

$$E_0^{f}(n) = E_0^{b}(n) = \lambda E_0^{f}(n-1) + x^2(n) \ , \quad E_{m-1}^{f}(n) = \lambda E_{m-1}^{f}(n-1) + \gamma_{m-1}(n-1) f_{m-1}^2(n) \ ,$$

$$E_{m-1}^{b}(n-1) = \lambda E_{m-1}^{b}(n-2) + \gamma_{m-1}(n-1) b_{m-1}^2(n-1) \ ,$$

$$f_m(n) = f_{m-1}(n) + k_m^{f}(n-1) b_{m-1}(n-1) \ ,$$

$$b_m(n) = b_{m-1}(n-1) + k_m^{b}(n-1) f_{m-1}(n) \ \left(m = 1 \ \text{to} \ N+1 \right)$$

$$k_m^{f}(n) = k_m^{f}(n-1) - \gamma_{m-1}(n-1) b_{m-1}(n-1) f_m(n) / E_{m-1}^{b}(n-1) \ ,$$

$$k_m^{b}(n) = k_m^{b}(n-1) - \gamma_{m-1}(n-1) f_{m-1}(n) b_m(n) / E_{m-1}^{f}(n) \ ,$$

$$\gamma_m(n-1) = \gamma_{m-1}(n-1) - \gamma_{m-1}^2(n-1) b_{m-1}^2(n-1) / E_{m-1}^{b}(n-1) \ ;$$

$$\left(n > 0 \ , \ m = 1, \cdots N+1 \right): \ e_m(n-1) = e_{m-1}(n-1) - w_{m-1}(n-2) b_{m-1}(n-1) \ ,$$

$$w_{m-1}(n-1) = w_{m-1}(n-2) + \gamma_{m-1}(n-1) b_{m-1}(n-1) e_m(n-1) / E_{m-1}^{b}(n-1) \ ,$$

$$e_m(n) = e_{m-1}(n) - w_{m-1}(n-1) b_{m-1}(n) \ .$$

Fig. 2.　Adaptive lattice ladder filter (ALLF) structure and the PEF-LSL algorithm [8].

its complexity is further reduced to $O(N)$. Fig. 2 shows the general structure of the adaptive lattice ladder filter (ALLF) and the prior error feedback-least square lattice (PEF-LSL) algorithms. More details of the PEF-LSL algorithm and other related algorithms can be found in the text books [8] and [23]–[25]. Like its RLS counterpart, the PEF-LSL algorithm is very sensitive to impulsive noise appearing in either $x(n)$ or $d(n)$, due to possible corrupt values of $f_0(n) = b_0(n) = x(n)$ and $e_0(n) = d(n)$ in the time and order recursions. It is therefore desirable to generalize the RLM algorithm introduced earlier to the PEF-LSL setting, where efficient order and time recursions can be utilized to reduce the arithmetic complexity.

Unfortunately, examination of (4) reveals that both the time- and order-recursive properties of $R_{X\rho}(n)$ are lost due to the in-

troduction of the nonlinear function $q(e(n))$. In other words, it is very difficult, if not impossible, to develop an exact time and order recursions similar to those in the LSL-like algorithms. Fortunately, we found that the M-estimate normal equation in (3) to (5) can be simplified further if $\rho(\cdot)$ is chosen as the modified Huber estimate function in (2). Of more importance still, it makes the order and time recursions feasible, which are almost identical to the conventional PEF-LSL algorithm. More precisely, when the input or desired signal is corrupted by impulses, $|e(n)|$ is likely to be larger than ξ, and $q(e(n))$ will be zero. In this case, (4) and (5) will be simplified to $R_{X\rho}(n) = \lambda R_{X\rho}(n-1)$ and $P_{X\rho}(n) = \lambda P_{X\rho}(n-1)$. In other words, $R_{X\rho}(n)$ and $P_{X\rho}(n)$ are simply multiplied by λ but not updated using the PEF-LSL iteration when impulses are detected.

TABLE I
HUBER PEF-LSL ALGORITHM

Initializations ($n=0$): $E_{0:N-1}^f(0) = E_{0:N-1}^b(-1) = \delta$, $\gamma_0(0) = 1$, $k_{1:N}^f(0) = k_{1:N}^b(0) = 0$,

$w_{0:N}(0) = 0$. $C_f = 1.483\left(1 + 5/(N_f - 1)\right)$, $C_e = 1.483\left(1 + 5/(N_e - 1)\right)$

Lattice prediction: ($n>0$, $m=1, \cdots N$), $\gamma_0(n-1) = 1$, $e_0(n) = d(n)$,

$\quad A_f(n) = [f_N^2(n), \cdots, f_N^2(n - N_1 + 1)]$, (if $n=1$, $A_f(1) = \mathbf{0}_{N_f \times 1}$)

$\quad \hat{\sigma}_f^2(n) = \lambda_\sigma \hat{\sigma}_f^2(n-1) + C_f(1 - \lambda_\sigma) \text{med}\left(A_f(n)\right)$, $\xi_f = k_\xi \hat{\sigma}_f(n)$,

if $\left|f_N(n)\right| > \xi_f$, then $x(n) = \sum_{m=1}^N b_m(n-1)k_m^f(n-1)$ end,

$\quad f_0(n) = b_0(n) = x(n)$, $E_0^f(n) = E_0^b(n) = \lambda E_0^f(n-1) + x^2(n)$,

$\quad E_{m-1}^f(n) = \lambda E_{m-1}^f(n-1) + \gamma_{m-1}(n-1)f_{m-1}^2(n)$,

$\quad E_{m-1}^b(n-1) = \lambda E_{m-1}^b(n-2) + \gamma_{m-1}(n-1)b_{m-1}^2(n-1)$,

$\quad f_m(n) = f_{m-1}(n) + k_m^f(n-1)b_{m-1}(n-1)$,

$\quad b_m(n) = b_{m-1}(n-1) + k_m^b(n-1)f_{m-1}(n)$ ($m=1$ to $N+1$),

$\quad k_m^f(n) = k_m^f(n-1) - \gamma_{m-1}(n-1)b_{m-1}(n-1)f_m(n)/E_{m-1}^b(n-1)$,

$\quad k_m^b(n) = k_m^b(n-1) - \gamma_{m-1}(n-1)f_{m-1}(n)b_m(n)/E_{m-1}^f(n)$,

$\quad \gamma_m(n-1) = \gamma_{m-1}(n-1) - \gamma_{m-1}^2(n-1)b_{m-1}^2(n-1)/E_{m-1}^b(n-1)$;

Filtering ($n>0$, $m=1, \cdots N+1$)

$\quad e_m(n-1) = e_{m-1}(n-1) - w_{m-1}(n-2)b_{m-1}(n-1)$,

$\quad w_{m-1}(n-1) = w_{m-1}(n-2) + \gamma_{m-1}(n-1)b_{m-1}(n-1)e_m(n-1)/E_{m-1}^b(n-1)$,

$\quad e_m(n) = e_{m-1}(n) - w_{m-1}(n-1)b_{m-1}(n)$,

$\quad A_e(n) = [e_{N+1}^2(n), \cdots, e_{N+1}^2(n - N_2 + 1)]$, (if $n=1$, $A_e(1) = \mathbf{0}_{N_e \times 1}$)

$\quad \hat{\sigma}_e^2(n) = \lambda_\sigma \hat{\sigma}_e^2(n-1) + C_e(1 - \lambda_\sigma) \text{med}\left(A_e(n)\right)$, $\xi_e = k_\xi \hat{\sigma}_e(n)$,

if $\left|e_{M+1}(n)\right| > \xi_e$, then $w_{m-1}(n-1) = w_{m-1}(n-2)$, $e_m(n) = \hat{\sigma}_e(n)$ end.

On the other hand, when no impulse is detected, $|e(n)|$ is less than ξ , and $q(e(n))$ is equal to one. Equations (4) and (5) are then identical to (12) and (13), and the order and time updates in the PEF-LSL algorithm will be performed as usual. In summary, the robust Huber PEF-LSL algorithm performs the normal order and time updates when no impulse is detected.

Another point worth mentioning is that due to the order recursion, the estimation error $e(n)$ will be available at the last stage of the ALLF (see Fig. 2) $e(n) = e_{N+1}(n)$. As mentioned earlier, if $|e(n)| > \xi$, then $d(n)$ or $x(n)$ are suspected to be corrupted by impulses. No updating will be performed, and the previous filtering parameters should be used instead of those generated in the current iteration. The case of input impulses, however, deserves further consideration. Although we can also detect this impulse from $e(n)$ and stabilize the filtering parameters $\boldsymbol{w}(n)$ by using their values in the previous iteration, its adverse effect on the lattice parameters cannot be effectively suppressed. Therefore, it is better to suppress the impulse in $x(n)$ before it enters the ALLF, using the method proposed in [13]. A similar idea has also been proposed by Kim et al. [14] in robust AR estimation under impulsive noise, where a prefilter based on linear prediction is used to remove the impulses in $x(n)$. In the proposed algorithm, an additional prefilter is not needed because such a prediction of $x(n)$ is already available from the lattice prediction part, similar to the robust gradient adaptive lattice filter in [13]. More precisely, a predictor for $x(n)$ can be formed as follows:

$$\hat{x}(n) = \sum_{m=1}^N b_m(n-1)k_m^f(n-1). \qquad (14)$$

If $|\hat{x}(n) - x(n)|$ is greater than a certain threshold ξ_x , then the input to the ALLF $x(n)$ will be replaced by $\hat{x}(n)$ to avoid possibly intrusion of the impulses. The selection of the threshold ξ_x can be done in the same way as described in (9) and (10). Details of the final H-PEF-LSL algorithm are summarized in Table I. This algorithm can be viewed as an effective implementation of the robust RLM algorithm, which preserves the advantages of the RLM and the lattice-based algorithms.

IV. MEAN AND MEAN SQUARE CONVERGENCE ANALYSIS

In this section, mean, and mean square behaviors of the RLM algorithm are evaluated under the following assumptions.

Assumption 1) The input signal $x(n)$ is ergodic with zero mean.

Assumption 2) The interference noise $\eta_o(n)$ is modeled as a contaminated Gaussian (CG) noise, which is a frequently

used model for analyzing impulsive interference [2], [4]. More specifically, the CG noise is given by

$$\eta_o(n) = \eta_g(n) + \eta_{im}(n) = \eta_g(n) + b(n)\eta_w(n) \qquad (15)$$

where $\eta_g(n)$ and $\eta_w(n)$ are independent zero mean Gaussian noises with variances σ_g^2 and σ_w^2, respectively; $b(n)$ is a switch sequence of ones and zeros, which is modeled as an independently and identically distributed (i.i.d) Bernoulli random process with occurrence probability $P_r(b(n) = 1) = p_r$ and $P_r(b(n) = 0) = 1 - p_r$. The variance of $\eta_w(n)$ is chosen to be much larger than that of $\eta_g(n)$ so that when $b(n) = 1$, a large impulse is experienced in $\eta_o(n)$. The ratio $r_{im} = p_r \cdot \sigma_w^2/\sigma_g^2 = \sigma_{im}^2/\sigma_g^2$ is used to determine the impulsive characteristic of $\eta_o(n)$, where σ_{im}^2 is the variance of the effective impulsive noise. For fixed value of σ_g^2, the larger the parameter r_{im}, the more impulsive the interference will be. The corresponding pdf of $\eta_o(n)$ in (15) is given by

$$f_{\eta_o}(\eta_o) = \frac{1 - p_r}{(2\pi\sigma_g^2)^{\frac{1}{2}}} \exp\left(-\frac{\eta_o^2}{2\sigma_g^2}\right) + \frac{p_r}{(2\pi\sigma_\Sigma^2)^{\frac{1}{2}}} \exp\left(-\frac{\eta_o^2}{2\sigma_\Sigma^2}\right)$$
$$(16)$$

where $\sigma_\Sigma^2 = \sigma_g^2 + \sigma_{im}^2 = \sigma_g^2 + p_r\sigma_w^2$ is the variance of $\eta_o(n)$.

Assumption 3) $x(n)$ and $e(n)$ have a jointly Gaussian distribution. $w(n)$, $x(n)$, and $\eta_o(n)$ are statistically independent. Although this assumption is not completely valid in general applications, it is commonly used to simplify the convergence analysis of a lot of adaptive filtering algorithms [2], [4], [14].

Assumption 4) $d_0(n) = (w^*)^T X(n)$, where w^* is the optimal Wiener solution given by $w^* = R_{E,X}^{-1} P_{E,Xd}$. $R_{E,X} \triangleq E[X(n)X^T(n)]$ is the ensemble averaged correlation matrix of $X(n)$, and $P_{E,dX} \triangleq E[d(n)X(n)]$ is the ensemble averaged cross-correlation vector between $X(n)$ and $d(n)$. Here, the subscript E indicates that the average is taken over the ensemble.

Assumption 5) $R_{E,X} = E[X(n)X^T(n)] \approx R_{X\rho}(n)/\Lambda(n)$ when n is large enough, where $\Lambda(n) = \sum_{i=1}^{n} \lambda_i^{(n)}$ is the normalization factor, and $R_{X\rho}(n)$ is estimated in (4).

Remarks of Assumption 5) It should be noted that when $X(i)$ is ergodic, the matrix $R_{E,X}$ can be approximated by $R_{E,X} \approx R_X(n)/\Lambda(n)$ [5], where $R_X(n) = \sum_{i=1}^{n} \lambda^{n-i} X(i)X^T(i)$ is the conventional time-averaged value [8], whereas, when there is impulsive noise, $R_{X\rho}(n)$ in (4) is a small biased approximation to $\Lambda(n)R_{E,X}(n)$ due to the use of $\rho(\cdot)$ and $\lambda_i^{(n)}$. An experiment was carried out to evaluate the effect of using $R_{X\rho}(n)$ in (4) when the desired signal is corrupted by a CG distributed interference under different impulsive noise density. For $p_r = 0.01$ to 0.1 and $\xi = 2.24\hat{\sigma}(n)$, it was found that the value $\|R_{X\rho} - \Lambda(n)R_{E,X}\|_2/\|R_{E,X}\|_2$ ($n = 1$ to 2048), averaged over 100 independent runs, is within 1.1 to 1.9%. This shows that the proposed estimator for $R_{X\rho}(n)$ in (4) is robust to different impulsive

noise density, and Assumption 5 is justified. This is also confirmed by the simulation results to be presented in Section V. Further theoretic results regarding the estimation error of $R_{X\rho}(n)$ can be found in [20]. Finally, it should be noted that if $X(n)$ is corrupted by impulsive noise, $R_{X\rho}(n)$ is a more robust estimator of the "impulse free" covariance matrix than $R_X(n)$.

Before proceeding to the convergence analysis, it should be noted that Weng and Leung [5] have presented convergence analysis for the N-RLS algorithm under the mixture of Gaussian noise. It was an extension of their previous works on Class A impulsive noise [22]. In their approach, the error sequence is assumed to be a zero mean Gaussian noise, which is based on the central limit theorem. Apart from the inherent differences of the RLM and N-RLS algorithms, our performance analysis differs from [5] in that we do not assume that the error sequence is Gaussian and instead rely on approximation (25), to be introduced in the sequel, in the analysis. This allows closed-form expressions for the mean and mean square analyzes to be obtained. At the same time, it also provides much insight into the behaviors of the RLM algorithm, as we will see later in this section. Finally, we remark that both approaches are based on the classical techniques in [17], which were originally developed for the performance analysis of LMS algorithm with Gaussian inputs.

For notational convenience, define $v(n) \triangleq w^* - w(n)$ as the weight-error vector. Substituting $w^* = R_{E,X}^{-1} P_{E,Xd}$ and $v(n)$ into (8), one gets

$$v(n) = v(n-1) - \frac{V(n-1)X(n)q(e(n))e(n)}{\lambda + q(e(n))X^T(n)V(n-1)X(n)} \qquad (17)$$

where $e(n) = d(n) - w^T(n-1)X(n) = \eta_o(n) + v^T(n-1)X(n)$. From Assumption 5, $V(n-1) = R_{X\rho}^{-1}(n-1)$ can be approximated by $\Lambda^{-1}(n-1)R_{E,X}^{-1}$, and (17) becomes

$$v(n) \approx v(n-1) - \frac{R_{E,X}^{-1}X(n)q(e(n))e(n)}{\bar{\lambda} + q(e(n))X^T(n)R_{E,X}^{-1}X(n)}$$
$$= v(n-1) - \frac{R_{E,X}^{-1}X(n)q(e(n))e(n)}{\bar{\lambda} + X^T(n)R_{E,X}^{-1}X(n)} \qquad (18)$$

where $\bar{\lambda} = \lambda\Lambda(n-1)$. The second equation in (18) is obtained by using the property of the weighting function $q(r) = \begin{cases} 1, & |r| \leq \xi \\ 0, & \text{otherwise} \end{cases}$.

A. Mean Behavior of the RLM Algorithm

Taking expectation over $\{v, X, \eta_o\}$ on both sides of (18), one gets

$$E[v(n)] = E[v(n-1)]$$
$$- R_{E,X}^{-1}\left\{E\left[\frac{X(n)q(e(n))e(n)}{\bar{\lambda} + X^T(n)R_{E,X}^{-1}X(n)}\right]\right\}$$
$$= E[v(n-1)] - R_{E,X}^{-1}L_1 \qquad (19)$$

where $E[\cdot]$ on the right-hand side of (19) denotes the expectation over $\{v(n-1), X(n), \eta_o(n)\}$, which is more clearly written

here as $E_{\{v,X,\eta_o\}}\{\cdot\}$. Dropping the time index from all variables, one gets

$$
\begin{aligned}
L_1 &\triangleq E_{\{v,X,\eta_o\}}\left[\frac{Xq(e)e}{\bar{\lambda}+X^T R_{E,X}^{-1} X}\right] \\
&= E_{\{v\}}\left\{E_{\{X,\eta_o\}}\left[\frac{Xq(e)e}{\bar{\lambda}+X^T R_{E,X}^{-1} X}\middle| v\right]\right\} \\
&= E_{\{v\}}[A] \tag{20}
\end{aligned}
$$

where the second equation is obtained from the independence assumption of $\eta_o(n)$, $w(n)$, and $x(n)$ in Assumption 3. The details for evaluating $A \triangleq E_{\{X,\eta_o\}}[Xq(e)e/(\bar{\lambda}+X^T R_{E,X}^{-1} X)|v]$ are given in Appendix A. From (A.14), we have

$$
\begin{aligned}
A &= \left\{C_B \iint_{N\text{ fold}} A_0(v,X)\exp\left(-\frac{1}{2}X^T B^{-1} X\right)dX\right\} \\
&\quad \times R_{E,X}v(n-1)\int_0^\infty \frac{\exp(-\beta\bar{\lambda})}{(2\beta+1)^{\frac{(N+2)}{2}}}d\beta \\
&= \left\{\frac{C_B}{2}\iint_{N\text{ fold}} A_0(v,X)\exp\left(-\frac{1}{2}X^T B^{-1} X\right)dX\right\} \\
&\quad \times \exp\left(\frac{\bar{\lambda}}{2}\right)E_{\frac{N}{2}+1}\left(\frac{\bar{\lambda}}{2}\right)R_{E,X}v(n-1) \tag{21}
\end{aligned}
$$

where $B = (1+2\beta)^{-1}R_{E,X}$, $C_B = (2\pi)^{-N/2}|B|^{-1/2}$, $E_k(x) = \int_1^\infty \exp(-x\beta)/\beta^k d\beta$ is the exponential integral function, and

$$
\begin{aligned}
A_0(v,X) &= \frac{1-p_r}{2}\left\{erf\left(\frac{\xi-v^T X}{\sqrt{2}\sigma_g}\right)+erf\left(\frac{\xi+v^T X}{\sqrt{2}\sigma_g}\right)\right\} \\
&\quad +\frac{p_r}{2}\left\{erf\left(\frac{\xi-v^T X}{\sqrt{2}\sigma_\Sigma}\right)+erf\left(\frac{\xi+v^T X}{\sqrt{2}\sigma_\Sigma}\right)\right\} \tag{22}
\end{aligned}
$$

which is bounded by $0 < A_0(v,X) \leq 2$. It is rather difficult to obtain a closed-form expression for the integral inside the $\{\cdot\}$ of (21). An approximation of $A_0(v,X)$ is therefore sought to simplify (21). First, it is noted that the threshold parameter ξ in (22) is estimated as $\xi = k_\xi \hat{\sigma}(n)$, where $\hat{\sigma}(n)$ is the standard deviation of the "impulse-free" estimation error. In the proposed algorithm, the impulsive component $\eta_{im}(n)$ of $\eta_o(n)$ is likely to be rejected, and the corresponding "impulse-free" estimation error can be approximated by $\hat{e}(n) \approx \eta_g(n) + v^T(n-1)X(n)$. Dropping the time index and using the condition that v, η_o, and X are uncorrelated, one gets

$$
\begin{aligned}
\xi &= k_\xi \sqrt{E[\hat{e}^2]} \\
&= k_\xi \left\{E[\eta_g^2] + E[(v^T X)^2]\right\}^{\frac{1}{2}} \\
&> k_\xi \left\{E[(v^T X)^2]\right\}^{\frac{1}{2}}. \tag{23}
\end{aligned}
$$

If $k_\xi \gg 1$, ξ will be much larger than $\{E[(v^T X)^2]\}^{1/2}$. Furthermore, if the probability density function of $v^T X$ decays sufficiently fast, it follows that

$$
P_r\left(\xi \gg |v^T X|\right) \approx 1 \tag{24}
$$

where P_r denotes the probability operator. Equation (24) is a consequence of the condition $\xi \gg \{E[(v^T X)^2]\}^{1/2}$ and the

Chernoff bound (see [15, p. 54]), although the latter bound is rather loose. This observation together with the property of the error function and the rapidly decaying exponential function $\exp(-(1/2)X^T B^{-1} X)$ in (21) allows us to use the following approximation of $A_0(v,X)$ in (22):

$$
A_0(v,X) \approx A_c = \left\{(1-p_r)erf\left(\frac{\xi}{\sqrt{2}\sigma_g}\right)+p_r erf\left(\frac{\xi}{\sqrt{2}\sigma_\Sigma}\right)\right\} \tag{25}
$$

during the evaluation of the expectation. Since $\sigma_\Sigma > \sigma_g$, A_c in (25) is bounded by $0 < A_c \leq 1$. From simulation results in Examples 3 and 4, it is found that A_c in (25) is a very good approximation to $A_0(v,X)$ under the specified simulation conditions. Inserting (25) into (21) yields

$$
A \approx A_c \exp\left(\frac{\bar{\lambda}}{2}\right)E_{\frac{N}{2}+1}\left(\frac{\bar{\lambda}}{2}\right)R_{E,X}v(n-1). \tag{26}
$$

Substituting (26) and (20) into (19), the following relation between $E[v(n)]$ and $E[v(n-1)]$ is obtained:

$$
E[v(n)] \approx \left\{1-A_c\exp\left(\frac{\bar{\lambda}}{2}\right)E_{\frac{N}{2}+1}\left(\frac{\bar{\lambda}}{2}\right)\right\}E[v(n-1)]. \tag{27}
$$

It can be verified that the term within the brace $\{\cdot\}$ in (27) is less than one because $0 < A_c \leq 1$ and $0 < \exp(\bar{\lambda}/2)$ $E_{N/2+1}(\bar{\lambda}/2) < 1$. The latter inequality results from the following properties of the function $E_k(x)$:

$$
\begin{aligned}
\frac{1}{x+k} &< \exp(x)E_k(x) < \frac{1}{x+k-1} \\
&\quad x > 0, \quad k = 1,2,\cdots \tag{28}
\end{aligned}
$$

$$
\begin{aligned}
\left(\frac{k}{k-1}\right)E_k(x) &< E_{k+1}(x) < E_k(x) \\
&\quad x > 0, \quad k = 1,2,\cdots \tag{29}
\end{aligned}
$$

and the fact that $\bar{\lambda}/2 > 0$ and $N/2+1 > 1$. Consequently, $\lim_{n\to\infty} E\{v(n)\} \approx 0$. This indicates that the RLM algorithm converges in the mean, and $w(n)$ converges approximately to the system parameter w^* under the stated assumptions. It should be noted that Assumption 3 does not apply to the conventional RLS algorithm when the impulse density p_r is high. The reason is that the impulses will change significantly the value of $w(n)$ and, hence, $v(n)$ in successive iterations. Therefore, $\eta_o(n)$, and $v(n)$ becomes correlated, whereas in the proposed RLM algorithm, the impulses will most likely be rejected, and $v(n)$ can be stabilized. Therefore, the independent assumption in Assumption 3 remains valid. In fact, if ξ is taken as infinity as in the conventional RLS algorithm, the impulses cannot be detected, and Assumption 3 is violated. The performance of the resulting algorithm will be inferior to one predicted here. Practical choices of ξ and their sensitivities are studied in Example 2. It was found that within a wide range of the threshold parameters, the RLM algorithm together with the robust parameter estimation method introduced in Section II is robust to impulsive noise. In addition, it can be seen from Example 3 that the theoretical result using the approximation in (27) agrees very well with the simulation results. This substantiates the validity of the proposed mean convergence analysis and theoretically explains the advantages of using the M-estimate cost function and the proposed
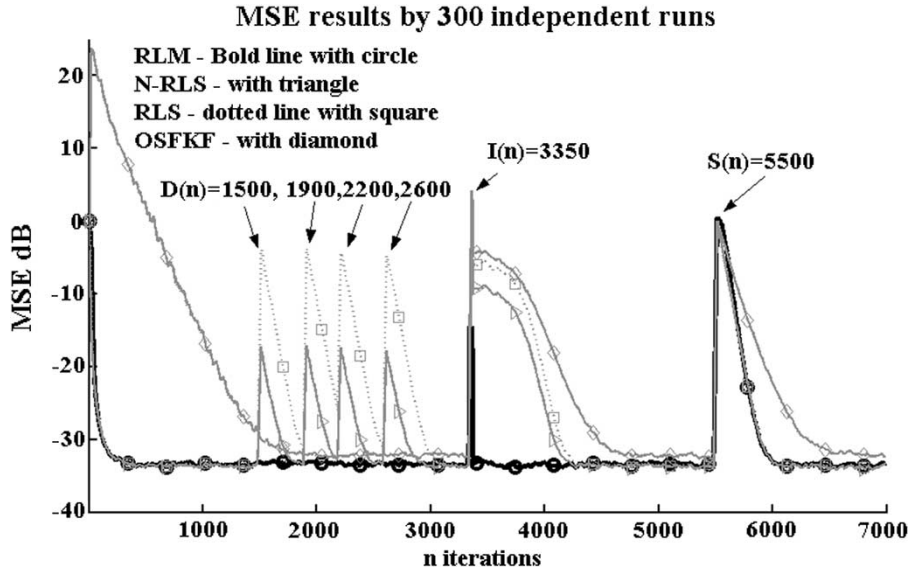
Fig. 3. MSE results of the various algorithms in the presence of the impulses (Example 1): RLS (dotted line with square), RLM (bold solid line with circle), N-RLS (solid line with triangle), and OSFKF (solid line with diamond). Impulses in the desired signal appear at $D(n) = 1500$, 1900, 2200, and 2600. Impulses in the input signal appear at $I(n) = 3350$. System abruptly changes at $S(n) = 5500$.

threshold estimation method over the conventional RLS algorithm. Next, we consider the mean square convergence of the RLM algorithm.

B. Mean Square Convergence of the RLM Algorithm

Postmultiplying (18) by its transpose and taking expectations on both sides over $\{v, X, \eta_o\}$ gives

$$\Xi(n) = \Xi(n-1) - R_{E,X}^{-1}S_1 - S_2 R_{E,X}^{-1} + R_{E,X}^{-1}S_3 R_{E,X}^{-1} \quad (30)$$

where $\Xi(n) \triangleq E[v(n)v^T(n)]$ is defined as the weight-error vector correlation matrix [8], and

$$S_1 = E_{\{v,X,\eta_o\}} \left\{ \frac{q(e)e X v^T}{\bar{\lambda} + X^T R_{E,X}^{-1} X} \right\}$$
$$= E_{\{v\}} \left\{ E_{\{X,\eta_o\}} \left[\frac{q(e)e X}{\bar{\lambda} + X^T R_{E,X}^{-1} X} \middle| v \right] v^T \right\}$$
$$= E_{\{v\}}[A v^T]$$
$$\approx \frac{A_c}{2} \exp\left(\frac{\bar{\lambda}}{2}\right) E_{\frac{N}{2}+1}\left(\frac{\bar{\lambda}}{2}\right) R_{E,X} \Xi(n-1) \quad (31)$$

$$S_2 = E_{\{v,X,\eta_o\}} \left\{ \frac{q(e)e v X^T}{\bar{\lambda} + X^T R_{E,X}^{-1} X} \right\}$$
$$= E_{\{v\}} \left\{ E_{\{X,\eta_o\}} v \left[\frac{q(e)e X^T}{\bar{\lambda} + X^T R_{E,X}^{-1} X} \middle| v \right] \right\}$$
$$= E_{\{v\}}\{v A^T\}$$
$$\approx \frac{A_c}{2} \exp\left(\frac{\bar{\lambda}}{2}\right) E_{\frac{N}{2}+1}\left(\frac{\bar{\lambda}}{2}\right) \Xi(n-1) R_{E,X} \quad (32)$$

where A is evaluated in Appendix A and is given in (21). Note that the approximation introduced in (26) has been used:

$$S_3 = E_{\{v,X,\eta_o\}} \left\{ \frac{q^2(e)e^2 X X^T}{\left(\bar{\lambda} + X^T R_{E,X}^{-1} X\right)^2} \right\}$$
$$= E_{\{v\}} \left\{ E_{\{X,\eta_o\}} \left[\frac{q^2(e)e^2 X X^T}{\left(\bar{\lambda} + X^T R_{E,X}^{-1} X\right)^2} \middle| v \right] \right\}$$
$$= E_{\{v\}}\{s_3\}$$
$$\approx \frac{1}{2} \exp\left(\frac{\bar{\lambda}}{2}\right) \left\{ E_{\frac{N+2}{2}}\left(\frac{\bar{\lambda}}{2}\right) - E_{\frac{N+4}{2}}\left(\frac{\bar{\lambda}}{2}\right) \right\}$$
$$\times A_c R_{E,X} \Xi(n-1) R_{E,X}$$
$$+ \frac{1}{4} \exp\left(\frac{\bar{\lambda}}{2}\right) \left\{ E_{\frac{N}{2}}\left(\frac{\bar{\lambda}}{2}\right) - E_{\frac{N+2}{2}}\left(\frac{\bar{\lambda}}{2}\right) \right\}$$
$$\times S_c R_{E,X} \quad (33)$$

where $s_3 = E_{\{X,\eta_o\}}[q^2(e)e^2 X X^T / (\bar{\lambda} + X^T R_{E,X}^{-1} X)^2 | v]$ is evaluated in Appendix B, and

$$S_c = -\frac{2(1-p_r)\sigma_g \xi}{\sqrt{2\pi}} \exp\left(-\frac{\xi^2}{2\sigma_g^2}\right) - \frac{2p_r \sigma_\Sigma \xi}{\sqrt{2\pi}} \exp\left(-\frac{\xi^2}{2\sigma_\Sigma^2}\right)$$
$$+ (1-p_r)\sigma_g^2 erf\left(\frac{\xi}{\sqrt{2}\sigma_g}\right) + p_r \sigma_\Sigma^2 erf\left(\frac{\xi}{\sqrt{2}\sigma_\Sigma}\right). \quad (34)$$

Substituting (31)–(33) into (30) yields

$$\Xi(n) \approx \Xi(n-1)\Gamma_1 + \Gamma_2 S_c R_{E,X}^{-1} \quad (35)$$

where

$$\Gamma_1 = 1 - \frac{A_c}{2}\left(\exp\left(\frac{\bar{\lambda}}{2}\right) E_{\frac{N}{2}+1}\left(\frac{\bar{\lambda}}{2}\right) + \exp\left(\frac{\bar{\lambda}}{2}\right) E_{\frac{N}{2}+2}\left(\frac{\bar{\lambda}}{2}\right)\right)$$
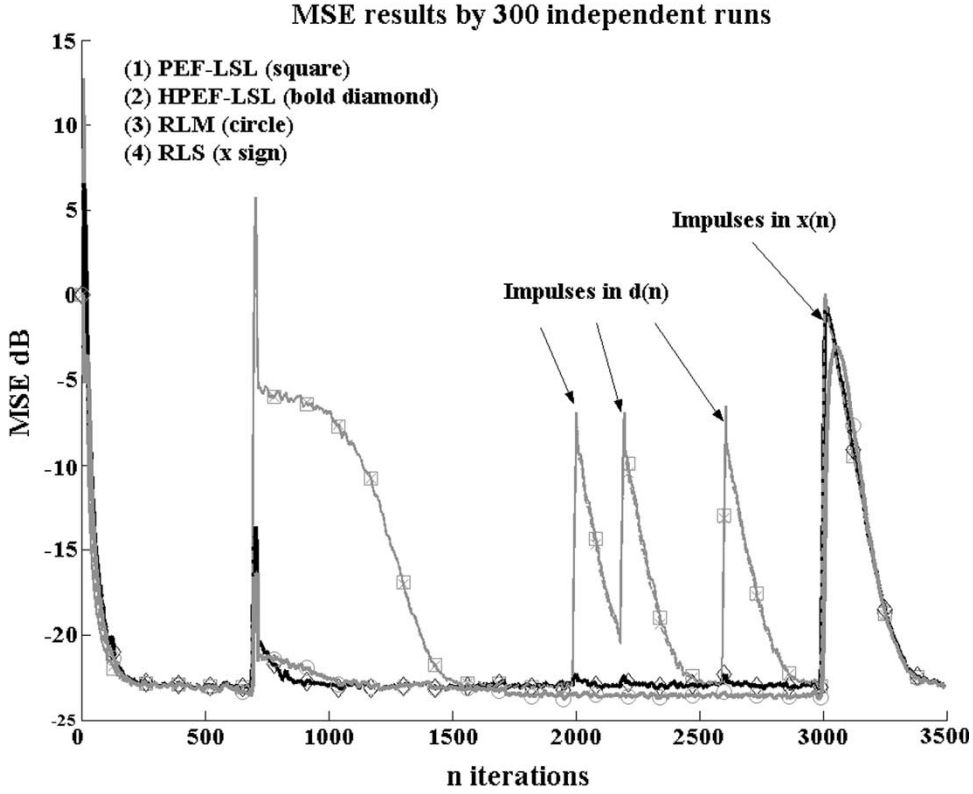$$(36)$$

Fig. 4. MSE performance of the various algorithms in impulsive noise (Example 1). (1) PEF-LSL (square); (2) H-PEF-LSL (bold diamond); (3) RLM (triangle); (4) RLS (x sign). $N = 9$, $\delta = 0.01$, $N_f = N_e = 5$ (for H-PEF-LSL), $\lambda = \lambda_\sigma = 0.99$, SNR $= 30$ dB, $\hat{\sigma}_e^2(0) = d^2(0)$, $\hat{\sigma}_f^2(0) = f_N^2(0)$, $N_w = 13$ (for RLM), $\mathbf{R}^{-1}(0) = \mathbf{I}$ (for RLS and RLM).

and

$$\Gamma_2 = \frac{1}{4} \exp\left(\frac{\bar{\lambda}}{2}\right) \left\{ E_{\frac{N}{2}}\left(\frac{\bar{\lambda}}{2}\right) - E_{\frac{N}{2}+1}\left(\frac{\bar{\lambda}}{2}\right) \right\}. \qquad (37)$$

It can be seen from (35) that $\boldsymbol{\Xi}(n)$ is prevented from going to zero by the last term. Postmultiplying $\boldsymbol{R}_{E,\boldsymbol{X}}$ and taking the trace operation on both sides of (35) yields

$$\mathbb{Z}(n) = \mathbb{Z}(n-1)\Gamma_1 + \Gamma_2 S_c \qquad (38)$$

where $\mathbb{Z}(n) \triangleq \text{trace}(\boldsymbol{\Xi}(n)\boldsymbol{R}_{E,\boldsymbol{X}})$ is defined as the misadjustment. Specifically, using the properties described in (28) and (29) with $0 < A_c < 1$, $\bar{\lambda} > 0$, and $N \gg 1$, the bound for Γ_1 from (36) can be deduced as $0 < \Gamma_1 < 1$. Similarly, with $\bar{\lambda} > 0$ and $N \gg 1$, Γ_2 from (37) is found to be a small positive value with bound: $0 < \Gamma_2 < (2/((\bar{\lambda} + N - 2)(\bar{\lambda} + N + 2)))$. Therefore, from (38), it can be seen that the weight-error vector converges in the mean square sense under the stated assumptions and that the steady-state error is mainly decided by the parameter S_c given in (34). Furthermore, the convergence of $\mathbb{Z}(n)$ is verified numerically by solving (38) under specified parameters, and the results are presented in Example 4 of Section V.

V. SIMULATION RESULTS

In order to evaluate the performance of the transversal RLM algorithm and the lattice-based H-PEF-LSL algorithm, the following simulations are performed on the system identification problem shown in Fig. 1 and Fig. 2 with impulsive interferences.

A. Example 1: Robustness and Convergence Performance of the RLM and the H-PEF-LSL Algorithms

This example compares the convergence speed and robustness of the RLM, N-RLS [5], OSFKF [3], and RLS algorithms with impulsive noise in the desired and input signals. The unknown system in Fig. 1(a) is modeled as a FIR filter with impulse response $\boldsymbol{w}^* = [0.2, -0.4, 0.6, -0.8, 1, -0.8, 0.6, -0.4, 0.2]^T$ and is changed suddenly to $-\boldsymbol{w}^*$ at time instant $n = 5500$ in order to evaluate the robustness of the algorithms to sudden change in system parameters. The input signal of the unknown system $x(n)$ is a colored signal with eigenvalue spread 46.821, which is generated by passing a zero-mean, unit variance, white Gaussian process through a linear time-invariant filter with coefficients [0.3887, 1, 0.3887]. These parameters are chosen for ease of comparison, and they are adopted from [8, pp. 581]. The signal-to-noise ratio (SNR) at the system output is given by SNR $= 10 \log_{10}(\sigma_{d_0}^2/\sigma_g^2)$, where $\sigma_{d_0}^2$ is the variance of the output of the unknown system. Unless otherwise specified, the following parameters will be used: Adaptive filter length N is 9, $r_{im} = 300$, SNR $= 35$ dB, the forgetting factors λ and λ_σ are 0.99, and the thresholds in (9) and (10) are used. The following initial values are used: $\hat{\sigma}^2(0) = d^2(0)$, $\boldsymbol{P}(0) = \mathbf{0}_{(N \times 1)}$, $\boldsymbol{w}(0) = \mathbf{0}_{(N \times 1)}$, and $\boldsymbol{V}(0) = 20\boldsymbol{I}_{(N \times N)}$, where $\mathbf{0}_{(I \times J)}$ and $\boldsymbol{I}_{(I \times J)}$ are the zero and identity matrices of dimension $(I \times J)$, respectively, and the constant 20 is chosen to ensure that $\boldsymbol{R}_{\boldsymbol{X}\rho}(0)$ consists of reasonable value. For illustration purposes, the noise $\eta_o(n)$ is an additive Gaussian noise from $n = 1$ to 1499 and 2801 to 7000 ($\eta_o(n) = \eta_g(n)$).
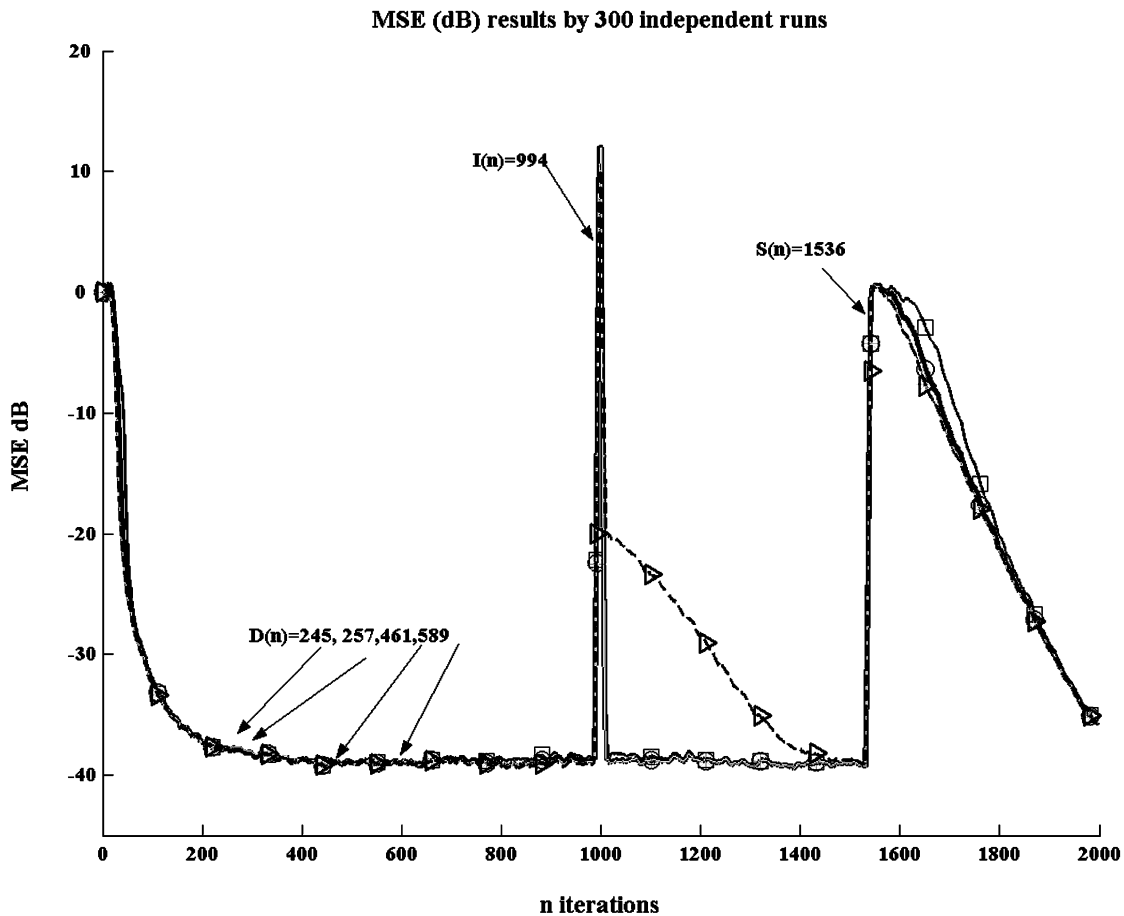
Fig. 5. MSE results for the RLM algorithm with different parameter thresholds k_ξ (Example 2). (1) $k_\xi = 1.2817$ (diamond); (2) $k_\xi = 1.96$ (circle); (3) $k_\xi = 2.576$ (star); (4) $k_\xi = 3.091$ (triangle).

From $n = 1500$ to 2800, a CG noise generated by (15) with $P_r(b(n) = 1) = p_r = 0.005$ is used. To visualize more clearly the effect of impulses in the desired signal, the appearing positions $D(\cdot)$ of impulses are fixed throughout this simulation; however, the amplitudes of impulses are independent random variables governed by (15). In addition, for simplicity in visualizing the effect of impulses in the input signal, one impulse is added to the input at $n = 3350$. The time instant for the system change $S(\cdot)$ is set as $n = 5500$. $\hat{\sigma}(n)$ is computed by (10), and the estimation window length N_w is chosen as 14 for the RLM and the OSFKF algorithms. The MSE results averaged over 300 independent runs are plotted in Fig. 3. It can be seen that the RLM, N-RLS, and the RLS algorithms have almost identical initial convergence speed, low steady-state error, and robustness to the system change. The RLS algorithm is not robust to any of the impulses, and the N-RLS algorithm is sensitive to impulses in the desired signal. It is clear that the performances of the N-RLS, RLS, and the OSFKF algorithms are deteriorated significantly by the impulse in the input signal at $n = 3350$. The impact of this impulse lasts for several hundred iterations. On the other hand, the RLM algorithm is able to recover very quickly from this adverse impact (about three times the filter length). The OSFKF algorithm can effectively suppress the impulses in the desired signal, but its convergence speed is slower, and the steady-state error is higher compared with other

RLS-type algorithms. Its robustness against the sudden system change is the worst. Comparisons with other robust LMS-like algorithms, such as ATNA, RMN, and the OSLMS algorithms, have also been performed. Due to space limitations, the results are not shown here. See [11] and [16] for more details. It is found that the RLM algorithm has faster initial convergence and lower steady-state error than those of the ATNA, RMN, and the OSLMS algorithms, due to its RLS-like nature. Moreover, the ATNA, RMN, and OSLMS algorithms are more susceptible to consecutive impulses either in the desired or the input signals. It should be noted that, because of the shifting property of the linear transversal filter, the effect of a single impulse at the input signal is similar to a series of consecutive impulses in the desired signal, which generates a sequence of estimation errors with large amplitudes. As the N-RLS algorithm only limits the effect of impulses instead of removing them completely, this algorithm is not robust to consecutive impulses. This also explains why it is not robust to the impulse in the input signal at $n = 3350$.

For clarity of presentation, the performance comparison of the proposed H-PEF-LSL, RLS, RLM, and PEF-LSL algorithms in an impulsive noise environment are separately plotted in Fig. 4, which is obtained by averaging the MSE results over 300 independent runs. From Fig. 4, we have the following observations.
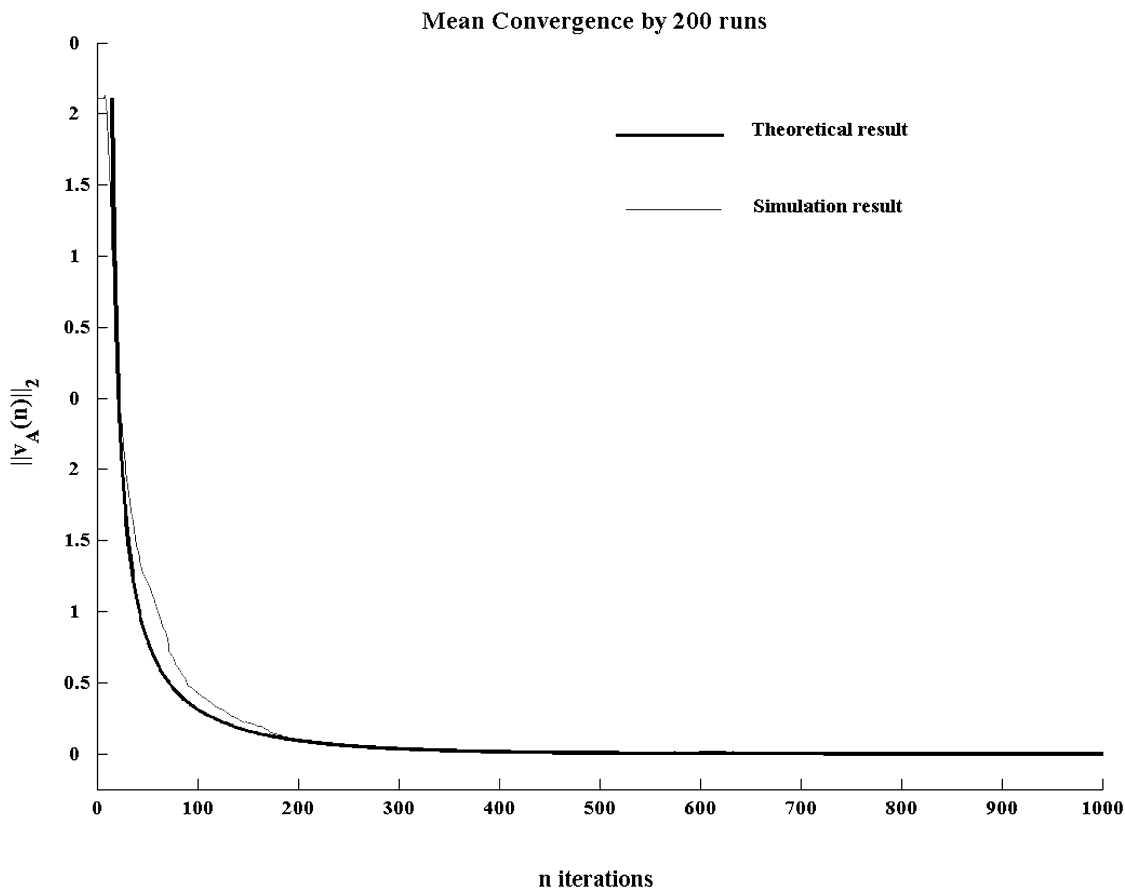
Fig. 6.   Mean convergence of the RLM algorithm by independent 200 runs (Example 3). The bold solid line is the theoretical result in (27) and the solid line is the simulation result.

1) The performances of the H-PEF-LSL and the RLM algorithms are very close to each other, with the latter being slightly better.
2) The performances of the RLS and the PEF-LSL algorithms are significantly degraded by the impulses, although the former is slightly better. The effect of a single impulse in $d(n)$ and $x(n)$ will last for more than 250 and 800 iterations for the RLS and the PEF-LSL algorithms, respectively.
3) The performance of the RLM and the H-PEF-LSL algorithms is very robust to the impulses in $d(n)$ and $x(n)$.
4) The initial convergence, steady state-error, and the tracking ability to sudden change of the system parameters of the H-PEF-LSL algorithm are comparable with other algorithms considered.

B. Example 2: Choice of the Thresholds

This experiment evaluates the performance of the transversal RLM algorithm with different choices of the threshold values ξ. The desired signal $d(n)$, the input signal $x(n)$, and additive interference are generated in the same way as in Example 1. The method proposed in (10) is used to estimate $\hat{\sigma}^2(n)$, and the threshold parameters are computed using (9) with different choices of the parameter k_ξ. Four different sets of threshold values with different confidence of impulse detection are used.

Specifically, the parameter k_ξ is chosen as 1.2817, 1.96, 2.576, and 3.091, respectively. This corresponds to $\theta_\xi = 0.2, 0.05, 0.01,$ and 0.005, respectively. The MSE for the RLM algorithm over 300 independent runs are plotted in Fig. 5. It can be seen that the performance of the RLM algorithm together with the parameter estimation method introduced in Section IV is robust to the impulsive noise in the desired signal within a wide range of threshold values. Moreover, it is also not sensitive to the choices of k_ξ when the input signal is corrupted by impulsive noise provided that they are not at the tails part of the signal distribution (such as $k_\xi = 3.091$ in Fig. 5), where impulse with large amplitude appears.

C. Example 3: Mean Convergence Performance of the RLM Algorithm

In this section, the mean convergence performance of the RLM algorithm will be evaluated. The norm of the mean squared weight-error vector is used as a performance measure

$$\|\boldsymbol{v}_A(n)\|_2 = \sqrt{\sum_{i=1}^{N} \left(\frac{1}{L} \sum_{j=1}^{L} v_i^{(j)}(n) \right)^2}$$
$$i = 1, \cdots, N, \ j = 1, \cdots, L \qquad (39)$$

where $v_i^{(j)}(n)$ is the ith component of the weight-error vector $\boldsymbol{v}(n)$ at time n in the jth independent run. L is the total number
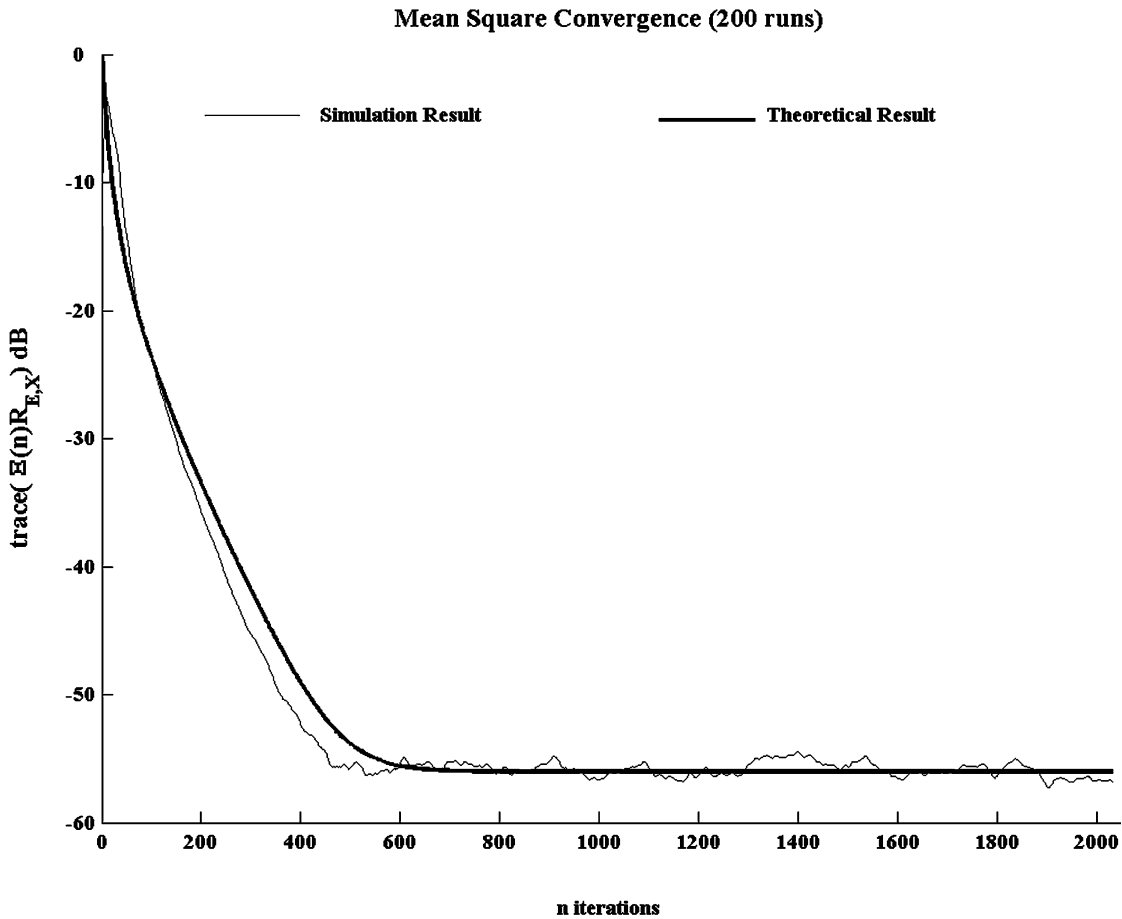
Fig. 7. Mean square convergence of the RLM algorithm by 200 independent runs (Example 4). The bold solid line is the theoretical result in (38), and the solid line is the simulation result.

of independent runs. In this example, the simulation result is obtained over $L = 200$ independent runs. The interference noise $\eta_o(n)$ is generated by (15) with $P(b(n) = 1) = p_r = 0.01$. However, the locations of impulses, unlike Example 1, are not fixed for each independent run. Only impulses in the desired signal are considered. The window length N_w is set to 9. The theoretical result is obtained from (27) with $\xi = 2.24 \cdot \tilde{\sigma}(n)$, where $\tilde{\sigma}(n)$ is obtained by averaging $\hat{\sigma}(n)$ over 200 independent runs. The resulting numerical and simulation results are plotted in Fig. 6. It can be seen that there is a good match between the theoretical and simulation results, especially when n is sufficiently large. The small discrepancy at the beginning may result from the inaccurate estimation of $\tilde{\sigma}(n)$ and the approximations used in the performance analysis.

D. Example 4: Mean Square Convergence Performance of the RLM Algorithm

This experiment follows the same settings in Example 3 to evaluate the mean square convergence performance of the RLM algorithm. The misadjustment $\mathbb{Z}(n) = \text{trace}(\Xi(n)\boldsymbol{R}_{E,\boldsymbol{X}})$ is considered. The numerical result obtained from (38) and the simulation results averaged over 200 independent runs are plotted in Fig. 7. Again, there is a good match between the theoretical and simulation results.

VI. CONCLUSION

This paper presents the convergence analysis of the RLM algorithm under the contaminated Gaussian (CG) impulsive noise model and a new lattice structure based fast algorithm, called the Huber Prior Error Feedback-Least Squares Lattice (H-PEF-LSL) algorithm, for its implementation. The H-PEF-LSL algorithm has an arithmetic complexity of $O(N)$ and is based on the modified Huber M-estimate function and the conventional PEF-LSL adaptive filtering algorithm. Simulation results show that the transversal RLM and the H-PEF-LSL algorithms have better performance than the conventional RLS and other RLS-like robust adaptive algorithms tested when the desired and input signals are corrupted by impulsive noise. Furthermore, the simulation and theoretical convergence results agree very well with each other and suggests that the RLM algorithm is more robust than the RLS algorithm under the CG noise model.

APPENDIX A

In this Appendix, the expectation $\boldsymbol{A} \triangleq E_{\{\boldsymbol{X},\eta_o\}}[\boldsymbol{X}q(e)e/(\bar{\lambda} + \boldsymbol{X}^T\boldsymbol{R}_{E,\boldsymbol{X}}^{-1}\boldsymbol{X})|\boldsymbol{v}]$ is evaluated. The subscript indicates that the expectation is taken over $\{\boldsymbol{X},\eta_o\}$. The classical approach proposed by Bershard in [17] is employed to evaluate this expecta-

tion. As $\eta_o(n)$ and $x(n)$ are assumed to be statistically independent, and $\boldsymbol{X}$ are jointly Gaussian with covariance matrix $\boldsymbol{R}_{E,\boldsymbol{X}}$, one gets

$$\boldsymbol{A} = C_R \iint\limits_{N+1 \text{ fold}} \frac{\boldsymbol{X}q(e)e}{\bar{\lambda} + \boldsymbol{X}^T \boldsymbol{R}_{E,\boldsymbol{X}}^{-1}\boldsymbol{X}}$$
$$\times \exp\left(-\frac{1}{2}\boldsymbol{X}^T \boldsymbol{R}_{E,\boldsymbol{X}}^{-1}\boldsymbol{X}\right) f_{\eta_o}(\eta_o)d\eta_o d\boldsymbol{X} \qquad \text{(A.1)}$$

where $C_R = (2\pi)^{-N/2}|\boldsymbol{R}_{E,\boldsymbol{X}}|^{-1/2}$. Similar to [17], let us consider the integral

$$\boldsymbol{F}(\beta) = C_R \iint\limits_{N+1 \text{ fold}} \frac{(\boldsymbol{X}q(e)e)\exp\left(-\beta\left(\bar{\lambda} + \boldsymbol{X}^T \boldsymbol{R}_{E,\boldsymbol{X}}^{-1}\boldsymbol{X}\right)\right)}{\bar{\lambda} + \boldsymbol{X}^T \boldsymbol{R}_{E,\boldsymbol{X}}^{-1}\boldsymbol{X}}$$
$$\times \exp\left(-\frac{\boldsymbol{X}^T \boldsymbol{R}_{E,\boldsymbol{X}}^{-1}\boldsymbol{X}}{2}\right) f_{\eta_o}(\eta_o)d\eta_o d\boldsymbol{X}. \qquad \text{(A.2)}$$

It can be seen that

$$\boldsymbol{A} = \boldsymbol{F}(0). \qquad \text{(A.3)}$$

Differentiating (A.2) with respect to β, one gets (A.4), shown at the bottom of the page, where $C_B = (2\pi)^{-N/2}|\boldsymbol{B}|^{-1/2}$, $\boldsymbol{B} = (1+2\beta)^{-1}\boldsymbol{R}_{E,\boldsymbol{X}}$, and $\boldsymbol{L}_2 \triangleq E_{\{\boldsymbol{X},\eta_o\}}[\boldsymbol{X}q(e)e|\boldsymbol{v}]|_{E[\boldsymbol{X}\boldsymbol{X}^T]=\boldsymbol{B}}$ is the expectation of $\boldsymbol{X}q(e)e$ conditioned on $\boldsymbol{v}$ when $x_i, x_j \in \boldsymbol{X}$ are jointly Gaussian with covariance matrix $\boldsymbol{B}$. Since $\boldsymbol{X}$ and e are assumed as jointly Gaussian in Assumption 3, the Price theorem [18] for $\boldsymbol{X}$ and e can be invoked to obtain the following:

$$\frac{\partial \boldsymbol{L}_2}{\partial \boldsymbol{r}_{E,Xe}} = E_{\{\boldsymbol{X},\eta_o\}}\left[\left.\left(\left(\frac{dq(e)}{de}e + q(e)\right)\right|\boldsymbol{v}\right)\right]\right|_{\boldsymbol{B}}$$
$$= E_{\{\boldsymbol{X},\eta_o\}}[q'(e)e|\boldsymbol{v}]\big|_{\boldsymbol{B}} + E_{\{\boldsymbol{X},\eta_o\}}[q(e)|\boldsymbol{v}]\big|_{\boldsymbol{B}}$$
$$= \boldsymbol{L}_{21} + \boldsymbol{L}_{22} \qquad \text{(A.5)}$$

where $q'(e) = dq(e)/de$, $\boldsymbol{r}_{E,Xe} \triangleq E_{\{\boldsymbol{X},\eta_o\}}[\boldsymbol{X}e|\boldsymbol{v}]|_{\boldsymbol{B}}$ is the covariance of $\boldsymbol{X}$ and e, $\boldsymbol{L}_{21} \triangleq E_{\{\boldsymbol{X},\eta_o\}}[q'(e)e|\boldsymbol{v}]|_{\boldsymbol{B}}$, and $\boldsymbol{L}_{22} \triangleq E_{\{\boldsymbol{X},\eta_o\}}[q(e)|\boldsymbol{v}]|_{\boldsymbol{B}}$. These expectations will be evaluated as follows. First of all, it is noted that

$$\boldsymbol{r}_{E,Xe} \triangleq E_{\{\boldsymbol{X},\eta_o\}}[\boldsymbol{X}e|\boldsymbol{v}]\big|_{\boldsymbol{B}}$$
$$= E_{\{\boldsymbol{X},\eta_o\}}\left[\boldsymbol{X}(\eta_o + \boldsymbol{X}^T\boldsymbol{v})|\boldsymbol{v}\right]\big|_{\boldsymbol{B}}$$
$$= \boldsymbol{B}\boldsymbol{v} = (1+2\beta)^{-1}\boldsymbol{R}_{E,\boldsymbol{X}}\boldsymbol{v}. \qquad \text{(A.6)}$$

Moreover, considering $\boldsymbol{X}$ and η_o are statistically independent, it follows that

$$\boldsymbol{L}_{21} = C_B \iint\limits_{N \text{ fold}} \left\{\int (q'(e)e)\, f_{\eta_o}(\eta_o)d\eta_o\right\}$$
$$\times \exp\left(-\frac{1}{2}\boldsymbol{X}^T\boldsymbol{B}^{-1}\boldsymbol{X}\right)d\boldsymbol{X}$$
$$= 0 \qquad \text{(A.7)}$$

because $\int (q'(e)e)f_{\eta_o}(\eta_o)d\eta_o$, as evaluated in the following, is an odd function:

$$\int (q'(e)e)\, f_{\eta_o}(\eta_o)d\eta_o$$
$$= \int \left[\delta(\eta_o + \boldsymbol{v}^T\boldsymbol{X} - \xi) - \delta(\eta_o + \boldsymbol{v}^T\boldsymbol{X} + \xi)\right]$$
$$\times (\eta_o + \boldsymbol{v}^T\boldsymbol{X})f_{\eta_o}(\eta_o)d\eta_o$$
$$= (\xi - \boldsymbol{v}^T\boldsymbol{X})f_{\eta_o}(\xi - \boldsymbol{v}^T\boldsymbol{X})$$
$$+ (-\xi - \boldsymbol{v}^T\boldsymbol{X})f_{\eta_o}(-\xi - \boldsymbol{v}^T\boldsymbol{X})$$
$$+ \boldsymbol{v}^T\boldsymbol{X}\left[f_{\eta_o}(\xi - \boldsymbol{v}^T\boldsymbol{X}) + f_{\eta_o}(-\xi - \boldsymbol{v}^T\boldsymbol{X})\right]$$
$$= \xi\left[f_{\eta_o}(\xi - \boldsymbol{v}^T\boldsymbol{X}) - f_{\eta_o}(-\xi - \boldsymbol{v}^T\boldsymbol{X})\right]. \qquad \text{(A.8)}$$

Here, we have used the identities $e = \eta_o + \boldsymbol{v}^T\boldsymbol{X}$ and $q'(e) = \delta(e + \xi) + \delta(e - \xi)$, where $\delta(\cdot)$ is the Dirac delta function. Equation (A.8) is an odd function because $f_{\eta_o}(\eta)$ is an even function. Next, we consider $\boldsymbol{L}_{22}$. Similar to deriving $\boldsymbol{L}_{21}$, we have

$$\boldsymbol{L}_{22} = C_B \iint\limits_{N \text{ fold}} \left\{\int q(e)f_{\eta_o}(\eta_o)d\eta_o\right\}$$
$$\times \exp\left(-\frac{1}{2}\boldsymbol{X}^T\boldsymbol{B}^{-1}\boldsymbol{X}\right)d\boldsymbol{X}$$
$$= C_B \iint\limits_{N \text{ fold}} \boldsymbol{A}_0(\boldsymbol{v},\boldsymbol{X})$$
$$\times \exp\left(-\frac{1}{2}\boldsymbol{X}^T\boldsymbol{B}^{-1}\boldsymbol{X}\right)d\boldsymbol{X} \qquad \text{(A.9)}$$

where $\boldsymbol{A}_0(\boldsymbol{v},\boldsymbol{X}) \triangleq \int q(e)f_{\eta_o}(\eta_o)d\eta_o$. Substituting $f_{\eta_o}(\eta_o)$ given in (16) into (A.9), one gets

$$A_0(\boldsymbol{v},\boldsymbol{X}) = \int_{-\xi}^{\xi} \frac{(1-p_r)}{\sqrt{2\pi}\sigma_g}\exp\left(-\frac{(e - \boldsymbol{v}^T\boldsymbol{X})^2}{2\sigma_g^2}\right)de$$
$$+ \int_{-\xi}^{\xi} \frac{p_r}{\sqrt{2\pi}\sigma_\Sigma}\exp\left(-\frac{(e - \boldsymbol{v}^T\boldsymbol{X})^2}{2\sigma_\Sigma^2}\right)de$$

$$\frac{d\boldsymbol{F}(\beta)}{d\beta} = -\frac{\exp(-\beta\bar{\lambda})}{(2\beta+1)^{\frac{N}{2}}}\left[C_B \iint\limits_{N+1 \text{ fold}} (\boldsymbol{X}q(e)e)\exp\left(-\boldsymbol{X}^T\left(\beta + \frac{1}{2}\right)\boldsymbol{R}_{E,\boldsymbol{X}}^{-1}\boldsymbol{X}\right)f_{\eta_o}(\eta_o)d\eta_o d\boldsymbol{X}\right]$$
$$= -\frac{\exp(-\beta\bar{\lambda})}{(2\beta+1)^{\frac{N}{2}}}E_{\{\boldsymbol{X},\eta_o\}}[\boldsymbol{X}q(e)e|\boldsymbol{v}]\bigg|_{E[\boldsymbol{X}\boldsymbol{X}^T]=\boldsymbol{B}} = -\frac{\exp(-\beta\bar{\lambda})}{(2\beta+1)^{\frac{N}{2}}}\boldsymbol{L}_2 \qquad \text{(A.4)}$$

$$
= (1 - p_r) \left\{ erf \left(\frac{\xi - \boldsymbol{v}^T \boldsymbol{X}}{\sqrt{2}\sigma_g} \right) \right.
$$

$$
\left. + erf \left(\frac{\xi + \boldsymbol{v}^T \boldsymbol{X}}{\sqrt{2}\sigma_g} \right) \right\}
$$

$$
+ p_r \left\{ erf \left(\frac{\xi - \boldsymbol{v}^T \boldsymbol{X}}{\sqrt{2}\sigma_\Sigma} \right) \right.
$$

$$
\left. + erf \left(\frac{\xi + \boldsymbol{v}^T \boldsymbol{X}}{\sqrt{2}\sigma_\Sigma} \right) \right\} \tag{A.10}
$$

where $\xi = k_\xi \hat{\sigma}(n)$, and $erf(x) = (2/\sqrt{\pi}) \int_0^x \exp(-t^2) dt$. Using the result in (A.7), (A.5) can be simplified to

$$
\frac{\partial \boldsymbol{L}_2}{\partial \boldsymbol{r}_{E,Xe}} = \boldsymbol{L}_{22} = C_B \iint_{N \text{ fold}} A_0(\boldsymbol{v}, \boldsymbol{X}) \exp\left(-\frac{1}{2} \boldsymbol{X}^T \boldsymbol{B}^{-1} \boldsymbol{X} \right) d\boldsymbol{X}. \tag{A.11}
$$

Integrating (A.11) with respect to $\boldsymbol{r}_{E,Xe}$ and using the expression of $\boldsymbol{r}_{E,Xe}$ in (A.6), one gets

$$
\boldsymbol{L}_2 = \left(C_{\boldsymbol{B}} \iint_{N \text{ fold}} A_0(\boldsymbol{v}, \boldsymbol{X}) \exp\left(-\frac{1}{2} \boldsymbol{X}^T \boldsymbol{B}^{-1} \boldsymbol{X} \right) d\boldsymbol{X} \right)
$$

$$
\times (1 + 2\beta)^{-1} \boldsymbol{R}_{E,\boldsymbol{X}} \boldsymbol{v}(n-1) + C_{\boldsymbol{L}_2} \tag{A.12}
$$

where $C_{\boldsymbol{L}_2} = E_{\{\boldsymbol{X},\eta_o\}}[\boldsymbol{X}q(e)e|\boldsymbol{v}]|\boldsymbol{r}_{E,Xe=0}$ is the constant of integration, which can be shown to be zero. Inserting (A.12) into (A.4) and integrating with respect to β yields

$$
\boldsymbol{F}(\beta) = - \int^\beta \frac{\exp(-\beta\bar{\lambda})}{(2\beta+1)^{\frac{(N+2)}{2}}} d\beta
$$

$$
\times \left(C_B \iint_{N \text{ fold}} A_0(\boldsymbol{v}, \boldsymbol{X}) \exp\left(-\frac{1}{2} \boldsymbol{X}^T \boldsymbol{B}^{-1} \boldsymbol{X} \right) d\boldsymbol{X} \right)
$$

$$
\times \boldsymbol{R}_{E,\boldsymbol{X}} \boldsymbol{v}(n-1) \tag{A.13}
$$

where the constant of integration is equal to zero because of the boundary condition $\boldsymbol{F}(\infty) = 0$. Finally, from (A.3) and (A.13), we have the desired result

$$
\boldsymbol{A} = \boldsymbol{F}(0)
$$

$$
= \left(C_B \iint_{N \text{ fold}} A_0(\boldsymbol{v}, \boldsymbol{X}) \exp\left(-\frac{1}{2} \boldsymbol{X}^T \boldsymbol{B}^{-1} \boldsymbol{X} \right) d\boldsymbol{X} \right)
$$

$$
\times \boldsymbol{R}_{E,\boldsymbol{X}} \boldsymbol{v}(n-1) \int_0^\infty \frac{\exp(-\beta\bar{\lambda})}{(2\beta+1)^{\frac{(N+2)}{2}}} d\beta. \tag{A.14}
$$

APPENDIX B

In this Appendix, $\boldsymbol{s}_3 = E_{\{\boldsymbol{X},\eta_o\}}[q^2(e)e^2 \boldsymbol{X}\boldsymbol{X}^T/(\bar{\lambda} + \boldsymbol{X}^T \boldsymbol{R}_{E,\boldsymbol{X}}^{-1} \boldsymbol{X})^2|\boldsymbol{v}]$ is evaluated. It is the expectation of $[q^2(e)e^2 \boldsymbol{X}\boldsymbol{X}^T/(\bar{\lambda} + \boldsymbol{X}^T \boldsymbol{R}_{E,\boldsymbol{X}}^{-1} \boldsymbol{X})^2]$ taken over $\{\boldsymbol{X},\eta_o\}$ conditioned on $\boldsymbol{v}$. Similar to deriving $\boldsymbol{A}$ in Appendix A, $\boldsymbol{s}_3$ is given by

$$
\boldsymbol{s}_3 = C_R \iint_{N+1 \text{ fold}} \frac{q^2(e)e^2 \boldsymbol{X}\boldsymbol{X}^T}{\left(\bar{\lambda} + \boldsymbol{X}^T \boldsymbol{R}_{E,\boldsymbol{X}}^{-1} \boldsymbol{X} \right)^2}
$$

$$
\times \exp\left(-\frac{1}{2} \boldsymbol{X}^T \boldsymbol{R}_{E,\boldsymbol{X}}^{-1} \boldsymbol{X} \right) f_{\eta_o}(\eta_o) d\eta_o d\boldsymbol{X} \tag{B.1}
$$

where $f_{\eta_o}(\eta_o)$ is the pdf of the contaminated Gaussian noise η_o given in (16). Similar to [17], let us define

$$
\bar{\boldsymbol{F}}(\beta) = C_R \iint_{N+1 \text{ fold}} \frac{\boldsymbol{X}\boldsymbol{X}^T q^2(e)e^2 \exp\left(-\beta\left(\bar{\lambda} + \boldsymbol{X}^T \boldsymbol{R}_{E,\boldsymbol{X}}^{-1} \boldsymbol{X} \right) \right)}{\left(\bar{\lambda} + \boldsymbol{X}^T \boldsymbol{R}_{E,\boldsymbol{X}}^{-1} \boldsymbol{X} \right)^2}
$$

$$
\times \exp\left(-\frac{\boldsymbol{X}^T \boldsymbol{R}_{E,\boldsymbol{X}}^{-1} \boldsymbol{X}}{2} \right) f_{\eta_o}(\eta_o) d\eta_o d\boldsymbol{X}. \tag{B.2}
$$

Comparing (B.2) with (B.1), it can be seen that

$$
\boldsymbol{s}_3 = \bar{\boldsymbol{F}}(0). \tag{B.3}
$$

To evaluate $\bar{\boldsymbol{F}}(\beta)$, differentiating (B.2) twice with respect to β, one gets (B.4), shown at the bottom of the page, where the correlation matrix $\boldsymbol{B}$ and C_B have been defined in Appendix A, and $\boldsymbol{L}_3 \triangleq E_{\{\boldsymbol{X},\eta_o\}}[q^2(e)e^2 \boldsymbol{X}\boldsymbol{X}^T|\boldsymbol{v}]|_{\boldsymbol{B}}$ is the expectation of

---

$$
\frac{d^2 \bar{\boldsymbol{F}}(\beta)}{d\beta^2} = C_R \exp(-\beta\bar{\lambda}) \iint_{N+1 \text{ fold}} \boldsymbol{X}\boldsymbol{X}^T q^2(e)e^2 \exp\left(-\boldsymbol{X}^T \left(\beta + \frac{1}{2} \right) \boldsymbol{R}_{E,\boldsymbol{X}}^{-1} \boldsymbol{X} \right) f_{\eta_o}(\eta_o) d\eta_o d\boldsymbol{X}
$$

$$
= \frac{\exp(-\beta\bar{\lambda})}{(2\beta+1)^{\frac{N}{2}}} \left[C_B \iint_{N+1 \text{ fold}} \boldsymbol{X}\boldsymbol{X}^T q^2(e)e^2 \exp\left(-\boldsymbol{X}^T \boldsymbol{B}^{-1} \boldsymbol{X} \right) f_{\eta_o}(\eta_o) d\eta_o d\boldsymbol{X} \right]
$$

$$
= \frac{\exp(-\beta\bar{\lambda})}{(2\beta+1)^{\frac{N}{2}}} E_{\{\boldsymbol{X},\eta_o\}} \left[q^2(e)e^2 \boldsymbol{X}\boldsymbol{X}^T|\boldsymbol{v} \right] \Big|_{E[\boldsymbol{X}\boldsymbol{X}^T]=\boldsymbol{B}}
$$

$$
= \frac{\exp(-\beta\bar{\lambda})}{(2\beta+1)^{\frac{N}{2}}} \boldsymbol{L}_3 \tag{B.4}
$$

$q^2(e)e^2\boldsymbol{XX}^T$ taken over $\{\boldsymbol{X}, \eta_o\}$ and conditioned on $\boldsymbol{v}$, where x_i, $x_j \in \boldsymbol{X}$ are jointly Gaussian variables with covariance matrix $\bar{\boldsymbol{B}}$. Following a similar approach used in [19], one gets, from the Price theorem, the following:

$$\frac{\partial \boldsymbol{L}_3}{\partial \boldsymbol{r}_{E,\boldsymbol{X}e}} = E_{\{\boldsymbol{X},\eta_o\}}\left[\left.\left(\frac{d\left[q^2(e)e^2\right]}{de}\boldsymbol{X}^T\right)\right|\boldsymbol{v}\right]\bigg|_{\boldsymbol{B}}$$
$$= 2E_{\{\boldsymbol{X},\eta_o\}}\left[\left.\left(q'(e)q(e)e^2\boldsymbol{X}^T + q^2(e)e\boldsymbol{X}^T\right)\right|\boldsymbol{v}\right]\bigg|_{\boldsymbol{B}}$$
$$= 2(\boldsymbol{L}_{31} + \boldsymbol{L}_{32}) \qquad (B.5)$$

where $\boldsymbol{L}_{31} \triangleq E_{\{\boldsymbol{X},\eta_o\}}[q'(e)q(e)e^2\boldsymbol{X}^T|\boldsymbol{v}]|_{\boldsymbol{B}}$, and $\boldsymbol{L}_{32} \triangleq E_{\{\boldsymbol{X},\eta_o\}}[q^2(e)e\boldsymbol{X}^T|\boldsymbol{v}]|_{\boldsymbol{B}}$. Again, applying a similar approach to derive $\boldsymbol{L}_{21}$ and $\boldsymbol{L}_{22}$ with the same argument in deriving (25), one gets

$$\boldsymbol{L}_{31} = C_B \iint_{N \text{ fold}} \left\{\int q'(e)q(e)e^2 f_{\eta_o}(\eta_o)d\eta_o\right\}\boldsymbol{X}^T$$
$$\times \exp\left(-\frac{1}{2}\boldsymbol{X}^T\boldsymbol{B}^{-1}\boldsymbol{X}\right)d\boldsymbol{X}$$
$$= C_B \iint_{N \text{ fold}} \left\{\xi^2\left[f_{\eta_o}(\xi-\boldsymbol{v}^T\boldsymbol{X})-f_{\eta_o}(\xi+\boldsymbol{v}^T\boldsymbol{X})\right]\right\}\boldsymbol{X}^T$$
$$\times \exp\left(-\frac{1}{2}\boldsymbol{X}^T\boldsymbol{B}^{-1}\boldsymbol{X}\right)d\boldsymbol{X} \approx 0 \qquad (B.6)$$

$$\boldsymbol{L}_{32} = C_B \iint_{N \text{ fold}} B_0(\boldsymbol{v},\boldsymbol{X})\boldsymbol{X}^T$$
$$\times \exp\left(-\frac{1}{2}\boldsymbol{X}^T\boldsymbol{B}^{-1}\boldsymbol{X}\right)d\boldsymbol{X} \qquad (B.7)$$

where $B_0(\boldsymbol{v},\boldsymbol{X}) \triangleq \int q^2(e)ef_{\eta_o}(\eta_o)d\eta_o$. Inserting the weighting function $q(r) = \begin{cases} 1, & |r| \le \xi \\ 0, & \text{otherwise} \end{cases}$, we have

$$B_0(\boldsymbol{v},\boldsymbol{X}) = \int_{-\xi}^{\xi} ef_{\eta_o}(\eta_o)d\eta_o$$
$$= -\frac{(1-p_r)\sigma_g}{\sqrt{2\pi}}\exp\left(-\frac{(\xi-\boldsymbol{v}^T\boldsymbol{X})^2}{2\sigma_g^2}\right)$$
$$+ \frac{(1-p_r)\boldsymbol{v}^T\boldsymbol{X}}{2}erf\left(\frac{\xi-\boldsymbol{v}^T\boldsymbol{X}}{\sqrt{2}\sigma_g}\right)$$
$$+ \frac{(1-p_r)\sigma_g}{\sqrt{2\pi}}\exp\left(-\frac{(\xi+\boldsymbol{v}^T\boldsymbol{X})^2}{2\sigma_g^2}\right)$$
$$+ \frac{(1-p_r)\boldsymbol{v}^T\boldsymbol{X}}{2}erf\left(\frac{\xi+\boldsymbol{v}^T\boldsymbol{X}}{\sqrt{2}\sigma_g}\right)$$
$$- \frac{p_r\sigma_\Sigma}{\sqrt{2\pi}}\exp\left(-\frac{(\xi-\boldsymbol{v}^T\boldsymbol{X})^2}{2\sigma_\Sigma^2}\right)$$
$$+ \frac{p_r\boldsymbol{v}^T\boldsymbol{X}}{2}erf\left(\frac{\xi-\boldsymbol{v}^T\boldsymbol{X}}{\sqrt{2}\sigma_\Sigma}\right)$$
$$+ \frac{p_r\sigma_\Sigma}{\sqrt{2\pi}}\exp\left(-\frac{(\xi+\boldsymbol{v}^T\boldsymbol{X})^2}{2\sigma_\Sigma^2}\right)$$
$$+ \frac{p_r\boldsymbol{v}^T\boldsymbol{X}}{2}erf\left(\frac{\xi+\boldsymbol{v}^T\boldsymbol{X}}{\sqrt{2}\sigma_\Sigma}\right). \qquad (B.8)$$

Following the same argument in deriving (25), (B.8) can also be approximated by

$$B_0(\boldsymbol{v},\boldsymbol{X}) \approx \boldsymbol{v}^T\boldsymbol{X}\left\{(1-p_r)erf\left(\frac{\xi}{\sqrt{2}\sigma_g}\right)+p_r erf\left(\frac{\xi}{\sqrt{2}\sigma_\Sigma}\right)\right\}$$
$$= A_c\boldsymbol{v}^T\boldsymbol{X}. \qquad (B.9)$$

Note that A_c is identical to that given in (25). Inserting (B.9) into (B.7) gives

$$\boldsymbol{L}_{32} \approx A_c\boldsymbol{v}^T\left(C_B \iint_{N \text{ fold}} \boldsymbol{XX}^T\exp\left(-\frac{1}{2}\boldsymbol{X}^T\boldsymbol{B}^{-1}\boldsymbol{X}\right)d\boldsymbol{X}\right)$$
$$= A_c\boldsymbol{v}^T\boldsymbol{B}. \qquad (B.10)$$

Substituting (B.6) and (B.10) into (B.5) yields

$$\frac{\partial \boldsymbol{L}_3}{\partial \boldsymbol{r}_{E,\boldsymbol{X}e}} \approx 2A_c\boldsymbol{v}^T\boldsymbol{B}. \qquad (B.11)$$

Integrating (B.11) with respect to $\boldsymbol{r}_{E,\boldsymbol{X}e}$ and using $\boldsymbol{r}_{E,\boldsymbol{X}e} = (1+2\beta)^{-1}\boldsymbol{R}_{E,\boldsymbol{X}}\boldsymbol{v}(n-1)$, one gets

$$\boldsymbol{L}_3 \approx 2A_c(1+2\beta)^{-2}\boldsymbol{R}_{E,\boldsymbol{X}}\boldsymbol{vv}^T\boldsymbol{R}_{E,\boldsymbol{X}} + C_{\boldsymbol{L}_3} \qquad (B.12)$$

where $C_{\boldsymbol{L}_3}$ is the integration constant, which can be determined from $E_{\{\boldsymbol{X},\eta_o\}}[q^2(e)e^2\boldsymbol{XX}^T|\boldsymbol{v}]$ conditioned on $\boldsymbol{r}_{E,\boldsymbol{X}e} = \boldsymbol{Bv} = 0$. In other words, the variables e and $\boldsymbol{X}$ are now uncorrelated. Then, we have

$$C_{\boldsymbol{L}_3} = E_{\{\boldsymbol{X},\eta_o\}}\left[q^2(e)e^2\boldsymbol{XX}^T\Big|\boldsymbol{v}\right]\bigg|_{\boldsymbol{r}_{E,\boldsymbol{X}e}=0}$$
$$= E_{\{\boldsymbol{X},\eta_o\}}\left[q^2(e)e^2\big|\boldsymbol{v}\right]\big|_{\boldsymbol{r}_{E,\boldsymbol{X}e}=0}\boldsymbol{B}$$
$$= \bar{C}_{\boldsymbol{L}_3}\big|_{\boldsymbol{r}_{E,\boldsymbol{X}e}=0}\boldsymbol{B}. \qquad (B.13)$$

where $\bar{C}_{\boldsymbol{L}_3} \triangleq E_{\{\boldsymbol{X},\eta_o\}}[q^2(e)e^2|\boldsymbol{v}]|_{\boldsymbol{B}}$. Following the definition of $\bar{C}_{\boldsymbol{L}_3}$, one gets

$$\bar{C}_{\boldsymbol{L}_3} = C_B \iint_{N \text{ fold}} C_0(\boldsymbol{v},\boldsymbol{X})\exp\left(-\frac{1}{2}\boldsymbol{X}^T\boldsymbol{B}^{-1}\boldsymbol{X}\right)d\boldsymbol{X}$$
$$(B.14)$$

where $C_0(\boldsymbol{v},\boldsymbol{X}) \triangleq \int q^2(e)e^2 f_{\eta_o}(\eta_o)d\eta_o$, which can be further simplified as follows:

$$C_0(\boldsymbol{v},\boldsymbol{X})$$
$$= \frac{1-p_r}{\sqrt{2\pi}\sigma_g}\int_{-\xi}^{\xi} e^2\exp\left(-\frac{(e-\boldsymbol{v}^T\boldsymbol{X})^2}{2\sigma_g^2}de\right)$$
$$+ \frac{p_r}{\sqrt{2\pi}\sigma_\Sigma}\int_{\xi}^{\xi} e^2\exp\left(-\frac{(e-\boldsymbol{v}^T\boldsymbol{X})^2}{2\sigma_\Sigma^2}\right)de$$
$$= \frac{(1-p_r)\sigma_g}{\sqrt{2\pi}}\left\{(-\xi+\boldsymbol{v}^T\boldsymbol{X})\exp\left(-\frac{(\xi+\boldsymbol{v}^T\boldsymbol{X})^2}{2\sigma_g^2}\right)\right.$$
$$\left.-(\xi+\boldsymbol{v}^T\boldsymbol{X})\exp\left(-\frac{(\xi-\boldsymbol{v}^T\boldsymbol{X})^2}{2\sigma_g^2}\right)\right\}$$
$$+(1-p_r)\left\{\left(\frac{\boldsymbol{v}^T\boldsymbol{XX}^T\boldsymbol{v}+\sigma_g^2}{2}\right)erf\left(\frac{\xi-\boldsymbol{v}^T\boldsymbol{X}}{\sqrt{2}\sigma_g}\right)\right.$$
$$\left.+\left\{\frac{\boldsymbol{v}^T\boldsymbol{XX}^T\boldsymbol{v}+\sigma_g^2}{2}\right\}erf\left(\frac{\xi+\boldsymbol{v}^T\boldsymbol{X}}{\sqrt{2}\sigma_g}\right)\right\}$$

$$+\frac{p_r\sigma_\Sigma}{\sqrt{2\pi}}\left\{(-\xi+\boldsymbol{v}^T\boldsymbol{X})\exp\left(-\frac{(\xi+\boldsymbol{v}^T\boldsymbol{X})^2}{2\sigma_\Sigma^2}\right)\right.$$
$$\left.-(\xi+\boldsymbol{v}^T\boldsymbol{X})\exp\left(-\frac{(\xi-\boldsymbol{v}^T\boldsymbol{X})^2}{2\sigma_\Sigma^2}\right)\right\}$$
$$+p_r\left\{\left(\frac{\boldsymbol{v}^T\boldsymbol{X}\boldsymbol{X}^T\boldsymbol{v}+\sigma_\Sigma^2}{2}\right)erf\left(\frac{\xi-\boldsymbol{v}^T\boldsymbol{X}}{\sqrt{2}\sigma_\Sigma}\right)\right.$$
$$\left.+\left(\frac{\boldsymbol{v}^T\boldsymbol{X}\boldsymbol{X}^T\boldsymbol{v}+\sigma_\Sigma^2}{2}\right)erf\left(\frac{\xi+\boldsymbol{v}^T\boldsymbol{X}}{\sqrt{2}\sigma_\Sigma}\right)\right\}. \quad (B.15)$$

Using the same argument in deriving (25), (B.15) can be approximated as

$$C_0(\boldsymbol{v},\boldsymbol{X}) \approx A_c\boldsymbol{v}^T\boldsymbol{X}\boldsymbol{X}^T\boldsymbol{v} + S_c \quad (B.16)$$

where A_c is given in (25), and

$$S_c = -\frac{2(1-p_r)\sigma_g\xi}{\sqrt{2\pi}}\exp\left(-\frac{\xi^2}{2\sigma_g^2}\right) - \frac{2p_r\sigma_\Sigma\xi}{\sqrt{2\pi}}\exp\left(-\frac{\xi^2}{2\sigma_\Sigma^2}\right)$$
$$+(1-p_r)\sigma_g^2 erf\left(\frac{\xi}{\sqrt{2}\sigma_g}\right) + p_r\sigma_\Sigma^2 erf\left(\frac{\xi}{\sqrt{2}\sigma_\Sigma}\right). \quad (B.17)$$

Inserting the result in (B.16) into (B.14) gives

$$\bar{C}_{\boldsymbol{L}_3} \approx C_B\iint\limits_{N\ \text{fold}}\{A_c\boldsymbol{v}^T\boldsymbol{X}\boldsymbol{X}^T\boldsymbol{v}+S_c\}\exp\left(-\frac{1}{2}\boldsymbol{X}^T\boldsymbol{B}^{-1}\boldsymbol{X}\right)d\boldsymbol{X}$$
$$= A_c\boldsymbol{v}^T\boldsymbol{B}\boldsymbol{v} + S_c. \quad (B.18)$$

Combining (B.15) and (B.18) yields the following:

$$C_{\boldsymbol{L}_3} = \bar{C}_{\boldsymbol{L}_3}\big|_{\boldsymbol{r}_E,\boldsymbol{X}_e=0}\boldsymbol{B} = S_c\boldsymbol{B} = S_c(1+2\beta)^{-1}\boldsymbol{R}_{E,\boldsymbol{X}}. \quad (B.19)$$

Finally, substituting (B.19) into (B.12), one gets

$$\boldsymbol{L}_3 \approx 2A_c(1+2\beta)^{-2}\boldsymbol{R}_{E,\boldsymbol{X}}\boldsymbol{v}\boldsymbol{v}^T\boldsymbol{R}_{E,\boldsymbol{X}} + S_c(1+2\beta)^{-1}\boldsymbol{R}_{E,\boldsymbol{X}}. \quad (B.20)$$

From (B.20) and (B.4), we have

$$\frac{d^2\bar{\boldsymbol{F}}(\beta)}{d\beta^2} = \frac{2\exp(-\beta\bar{\lambda})}{(2\beta+1)^{\frac{(N+4)}{2}}}A_c\boldsymbol{R}_{E,\boldsymbol{X}}\boldsymbol{v}\boldsymbol{v}^T\boldsymbol{R}_{E,\boldsymbol{X}}$$
$$+\frac{\exp(-\beta\bar{\lambda})}{(2\beta+1)^{\frac{(N+2)}{2}}}S_c\boldsymbol{R}_{E,\boldsymbol{X}}. \quad (B.21)$$

Integrating (B.21) with respect to β yields

$$\boldsymbol{s}_3 = \bar{\boldsymbol{F}}(0)$$
$$= \int_0^\infty\int_{\beta_1}^\infty\frac{2A_c\boldsymbol{R}_{E,\boldsymbol{X}}\boldsymbol{v}\boldsymbol{v}^T\boldsymbol{R}_{E,\boldsymbol{X}}\exp(-\beta_2\bar{\lambda})}{(2\beta_2+1)^{\frac{(N+4)}{2}}}d\beta_2 d\beta_1$$
$$+\int_0^\infty\int_{\beta_1}^\infty\frac{S_c\boldsymbol{R}_{E,\boldsymbol{X}}\exp(-\beta_2\bar{\lambda})}{(2\beta_2+1)^{\frac{(N+2)}{2}}}d\beta_2 d\beta_1$$
$$= \frac{1}{2}\exp\left(\frac{\bar{\lambda}}{2}\right)\left\{E_{\frac{N+2}{2}}\left(\frac{\bar{\lambda}}{2}\right) - E_{\frac{N+4}{2}}\left(\frac{\bar{\lambda}}{2}\right)\right\}$$
$$\times A_c\boldsymbol{R}_{E,\boldsymbol{X}}\boldsymbol{v}\boldsymbol{v}^T\boldsymbol{R}_{E,\boldsymbol{X}}$$
$$+\frac{1}{4}\exp\left(\frac{\bar{\lambda}}{2}\right)\left\{E_{\frac{N}{2}}\left(\frac{\bar{\lambda}}{2}\right) - E_{\frac{N+2}{2}}\left(\frac{\bar{\lambda}}{2}\right)\right\}$$
$$\times S_c\boldsymbol{R}_{E,\boldsymbol{X}} \quad (B.22)$$

with the boundary conditions $\bar{F}(\infty) = 0$ and $\partial\bar{F}(\beta)/\partial\beta|_{\beta=\infty} = 0$. Equation (B.22) is obtained from (B.3) and

the explicitly evaluation of the integrations. Actually, the integrations in (B.22) are double integral in the first half of the first quadrant in the β_1, β_2 plane. Interchanging the order of integration [17], we have

$$I_1 = \int_0^\infty\int_{\beta_1}^\infty\frac{\exp(-\beta_2\bar{\lambda})}{(2\beta_2+1)^{\frac{(N+4)}{2}}}d\beta_2 d\beta_1$$
$$= \int_0^\infty\int_0^{\beta_2}\frac{\exp(-\beta_2\bar{\lambda})}{(2\beta_2+1)^{\frac{(N+4)}{2}}}d\beta_1 d\beta_2$$
$$= \int_0^\infty\frac{\beta_2\exp(-\beta_2\bar{\lambda})}{(2\beta_2+1)^{\frac{(N+4)}{2}}}d\beta_2$$
$$= \frac{1}{4}\exp\left(\frac{\bar{\lambda}}{2}\right)\left\{E_{\frac{N+2}{2}}\left(\frac{\bar{\lambda}}{2}\right) - E_{\frac{N+4}{2}}\left(\frac{\bar{\lambda}}{2}\right)\right\} \quad (B.23)$$

where $E_k(x) = \int_1^\infty\exp(-x\bar{\beta})/\bar{\beta}^k d\beta$. Similarly

$$I_2 \triangleq \int_0^\infty\int_{\beta_1}^\infty\frac{\exp(-\beta_2\bar{\lambda})}{(2\beta_2+1)^{\frac{(N+2)}{2}}}d\beta_2 d\beta_1$$
$$= \frac{1}{4}\exp\left(\frac{\bar{\lambda}}{2}\right)\left\{E_{\frac{N}{2}}\left(\frac{\bar{\lambda}}{2}\right) - E_{\frac{N+2}{2}}\left(\frac{\bar{\lambda}}{2}\right)\right\}. \quad (B.24)$$

ACKNOWLEDGMENT

## REFERENCES

[1] K. L. Blackard, T. S. Rappaport, and C. W. Bostian, "Measurements and models of radio frequency impulsive noise for indoor wireless communications," *IEEE J. Select. Areas Commun.*, vol. 11, pp. 991–1001, Sept. 1993.

[2] T. I. Haweel and P. M. Clarkson, "A class of order statistic LMS algorithms," *IEEE Trans. Signal Processing*, vol. 40, pp. 44–53, Jan. 1992.

[3] R. Settineri, M. Najim, and D. Ottaviani, "Order statistic fast Kalman filter," in *Proc. IEEE Int. Symp. Circuits Syst.*, vol. 2, 1996, pp. 116–119.

[4] S. Koike, "Adaptive threshold nonlinear algorithm for adaptive filters with robustness against impulsive noise," *IEEE Trans. Signal Processing*, vol. 45, pp. 2391–2395, Sept. 1997.

[5] J. F. Weng and S. H. Leung, "Adaptive nonlinear RLS algorithm for robust filtering in impulse noise," in *Proc. IEEE Int. Symp. Circuits Syst.*, vol. 4, 1997, pp. 2337–2340.

[6] J. A. Chambers and A. Avlonitis, "A Robust Mixed-Norm (RMN) adaptive filter algorithm," *IEEE Signal Processing Lett.*, vol. 4, pp. 46–48, Feb. 1997.

[7] Y. Zou, S. C. Chan, and T. S. Ng, "A Recursive Least M-Estimate (RLM) Adaptive filter for robust filtering in impulsive noise," *IEEE Signal Processing Lett.*, vol. 7, pp. 324–326, Nov. 2000.

[8] S. S. Haykin, *Adaptive Filter Theory*, 3rd ed. Englewood Cliffs, N.J.: Prentice-Hall, 1996.

[9] F. R. Hampel, E. M. Ronchetti, P. J. Rousseeuw, and W. A. Stahel, *Robust Statistics: The Approach Based on Influence Functions*. New York: Wiley, 1986.

[10] P. J. Huber, *Robust Statistics*. New York: Wiley, 1981.

[11] Y. Zou, "Robust Statistics Based Adaptive Filtering Algorithms for Impulsive Noise Suppression," Ph.D. Dissertation, The Univ. Hong Kong, Hong Kong, 2000.

[12] P. J. Rousseeuw and A. M. Leroy, *Robust Regression and Outlier Detection*. New York: Wiley, 1987.

[13] Y. Zou, S. C. Chan, and T. S. Ng, "A robust statistics based adaptive lattice-ladder filter in impulsive noise," in *Proc. IEEE Int. Symp. Circuits Syst.*, vol. 3, 2000, pp. 539–542.

[14] S. R. Kim and A. Efron, "Adaptive robust impulsive noise filtering," *IEEE Trans. Signal Processing*, vol. 43, pp. 1855–1866, Aug. 1995.

[15] J. G. Proakis, *Digital Communications*, 3rd ed. New York: McGraw-Hill, 1995.

[16] Y. Zou, S. C. Chan, and T. S. Ng, "Least mean M-estimate algorithms for robust adaptive filtering in impulse noise," *IEEE Trans. Circuits Syst. II*, vol. 47, pp. 1564–1569, Dec. 2000.

[17] N. J. Bershard, "Analysis of the normalized LMS algorithm with Guassian inputs," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-34, pp. 793–806, Apr. 1986.

[18] A. Papoulis, *Probability, Random Variables, and Stochastic Processes*, 3rd ed. New York: McGraw-Hill, 1991.

[19] V. J. Mathews, "Performance analysis of adaptive filters equipped with the dual sign algorithm," *IEEE Trans. Signal Processing*, vol. 39, pp. 85–91, Jan. 1991.

[20] C. J. Masreliez and R. D. Martin, "Robust Bayesian estimation for the linear model and robustifying the Kalman filter," *IEEE Trans. Automat. Contr.*, vol. AC-22, pp. 361–371, June 1977.

[21] D. Middleton, "Man-made noise in urban environments and transportation systems: models and measurements," *IEEE Trans. Commun.*, vol. COM-21, pp. 1232–1241, Nov. 1973.

[22] J. F. Weng, "Performance of DS/CDMa two-stage nonlinear detector in impulse noise," Ph.D. dissertation, City Univ. Hong Kong, Hong Kong, 1997.

[23] J. G. Proakis, C. M. Rader, F. Ling, and C. L. Nikias, *Advanced Digital Signal Processing*. New York: Macmillan, 1992.

[24] J. G. Proakis, C. M. Rader, F. Ling, M. Moonen, I. K. Proudler, and C. L. Nikias, *Algorithms for Statistical Signal Processing*. Englewood Cliffs, NJ: Prentice-Hall, 2002.

[25] D. G. Manolakis, V. K. Ingle, and S. M. Kogon, *Statistical and Adaptive Signal Processing: Spectral Estimation, Signal Modeling, Adaptive Filtering, and Array Processing*. New York: McGraw-Hill, 2000.
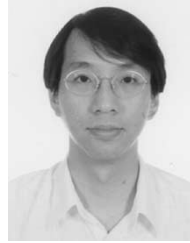
Shing-Chow Chan (S'87-M'92) received the B.Sc. (Eng) and Ph.D. degrees from the University of Hong Kong in 1986 and 1992, respectively.

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 an associate Professor. He was a visiting researcher in Microsoft Corporation, Redmond, WA, and Microsoft China in 1998 and 1999, respectively. His research interests include fast transform algorithms, filter design and realization, multirate signal processing, communications signal processing, and image-based rendering.

Dr. Chan is currently a member of the Digital Signal Processing Technical Committee of the IEEE Circuits and Systems Society. He was Chairman of the IEEE Hong Kong Chapter of Signal Processing from 2000 to 2002.

Yue-Xian Zou received the M.Sc. degree from the University of Electronic Science and Technology of China, ChengDu, China, in 1991 and the Ph.D. degree from the University of Hong Kong in 2001.

She is a lecturer at Singapore Polytechnic. She is currently working on a lung sound analysis project. Her research interests include adaptive signal processing and its applications: higher order statistics, wavelets, polyspectra, feature extraction, neural networks, and pattern recognition for biomedical signal and image processing.