Robust Recursive Least-Squares Adaptive-Filtering Algorithm for Impulsive-Noise Environments

Md. Zulfiquar Ali Bhotto, *Student Member, IEEE*, and Andreas Antoniou, *Life Fellow, IEEE*

***Abstract*—A new robust recursive least-squares (RLS) adaptive filtering algorithm that uses *a priori* error-dependent weights is proposed. Robustness against impulsive noise is achieved by choosing the weights on the basis of the L_1 norms of the cross-correlation vector and the input-signal autocorrelation matrix. The proposed algorithm also uses a variable forgetting factor that leads to fast tracking. Simulation results show that the proposed algorithm offers improved robustness as well as better tracking compared to the conventional RLS and recursive least-M estimate adaptation algorithms.**

***Index Terms*—Adaptive filters, RLS adaptation algorithms, robust adaptation algorithms.**

I. Introduction

The convergence performance of adaptive filters depends critically on the randomness of the input-desired signal pairs [1]. Impulsive disturbances or noise in the input-desired signal pairs can cause the performance of adaptive filters to deteriorate [2]. Robustness in adaptive filters in impulsive-noise environments is achieved in a number of ways [2]. In [3]–[5], adaptive-filter robustness is considered as insensitivity to impulsive noise and in [6] it is deemed to be the capability of an adaptive filter to reconverge to the steady-state solution at the same rate of convergence as before. The robust algorithms in [3], [4] use the Hampel three-part redescending M-estimate objective function and that in [5] uses the Huber two-part M-estimate objective function. In [3]–[5], the median absolute deviation (MAD) [7] is used to estimate the variance of the error signal in order to determine appropriate threshold values. The amplitude of the error signal is then compared with these thresholds values to detect the presence of impulsive noise and whenever such noise is present, the algorithm either reduces the learning rate significantly or discards the error signal completely in the coefficient-vector update equation. In [6], the instantaneous power of the weighted error signal is lowpass filtered and then used to switch the step size of the algorithm between two levels one of which suppresses the error signal corrupted by impulsive noise during the adaptation of the coefficient vector. The robust algorithms in [4]–[6] belong to the recursive least-squares (RLS) family and hence they converge significantly faster than algorithms of the steepest-descent family [1].

Manuscript received December 01, 2010; revised January 06, 2011; accepted January 06, 2011. Date of publication January 13, 2011; date of current version January 27, 2011. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Hsiao-Chun Wu.

The authors are with the Department of Electrical and Computer Engineering, University of Victoria, Victoria, BC V8W 3P6 Canada (e-mail: zbhotto@ece.uvic.ca; aantoniou@ieee.org).

Colors version of one or more figures in this paper are available at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/LSP.2011.2106119

In this paper, we propose a new robust RLS adaptive-filtering algorithm that yields an optimal solution of the weighted least-squares optimization problem. The proposed algorithm is robust with respect to impulsive noise as well as long bursts of impulsive noise in the sense that it converges back to the steady state much faster than during the initial convergence. The proposed algorithm also tracks sudden system disturbances. Simulation results show that the proposed algorithm achieves improved robustness and better tracking as compared to the conventional RLS and recursive least-M estimate (RLM) algorithms reported in [4]. The paper is organized as follows. In Section II, the proposed robust RLS algorithm is described. In Section III simulation results are presented and finally conclusions are drawn in Section IV.

II. Proposed Robust RLS Algorithm

Two slightly different versions of the proposed robust RLS algorithm are possible as detailed below, one for stationary and the other for nonstationary environments.

A. Robust RLS Algorithm for Stationary Environments

Weighted least-squares algorithms obtain the optimal coefficient vector $\boldsymbol{w}_k$ at iteration k by solving the optimization problem

$$\underset{\boldsymbol{w}_k}{\text{minimize}} \sum_{i=1}^{k} q_i \left(d_i - \boldsymbol{w}_k^T \boldsymbol{x}_i\right)^2 \tag{1}$$

where d_i is the desired signal, $\boldsymbol{x}_i$ is the input signal vector, and q_i is a nonnegative weight at iteration i. Each of vectors $\boldsymbol{w}_k$ and $\boldsymbol{x}_i$ is of dimension M. The solution of (1) is achieved by solving the normal equations which are obtained by setting the gradient of the objective function in (1) with respect to $\boldsymbol{w}_k$ to zero. The input-signal autocorrelation matrix, $\boldsymbol{R}_k$, and crosscorrelation vector, $\boldsymbol{p}_k$, at iteration k are given by

$$\boldsymbol{R}_k = \lambda_f \boldsymbol{R}_{k-1} + \delta_k \boldsymbol{x}_k \boldsymbol{x}_k^T \tag{2}$$

$$\boldsymbol{p}_k = \lambda_f \boldsymbol{p}_{k-1} + \delta_k \boldsymbol{x}_k d_k \tag{3}$$

where $\boldsymbol{R}_k$ and $\boldsymbol{p}_k$ are of dimensions $[M, M]$ and M, respectively, and $0 \ll \lambda_f < 1$. Parameter λ_f is a prespecified fixed forgetting factor and δ_k is a nonnegative scalar. The normal equations of (1) can be expressed in matrix form as

$$\boldsymbol{R}_k \boldsymbol{w}_k = \boldsymbol{p}_k \tag{4}$$

Using the matrix inversion lemma [1], [2] in (2), we obtain the update equation of the inverse of the autocorrelation matrix as

$$\boldsymbol{S}_k = \frac{1}{\lambda_f} \left(\boldsymbol{S}_{k-1} - \frac{1}{\dfrac{\lambda_f}{\delta_k} + \boldsymbol{x}_k^T \boldsymbol{S}_{k-1} \boldsymbol{x}_k} \boldsymbol{S}_{k-1} \boldsymbol{x}_k \boldsymbol{x}_k^T \boldsymbol{S}_{k-1} \right). \tag{5}$$

Now using (5) in (4), the update equation of the coefficient vector is obtained as

$$\boldsymbol{w}_k = \boldsymbol{w}_{k-1} + \frac{1}{\frac{\lambda_f}{\delta_k} + \boldsymbol{x}_k^T \boldsymbol{S}_{k-1}\boldsymbol{x}_k} \boldsymbol{S}_{k-1}\boldsymbol{x}_k e_k \tag{6}$$

where

$$e_k = d_k - \boldsymbol{w}_{k-1}^T \boldsymbol{x}_k \tag{7}$$

is the *a priori* error. In impulsive-noise environments, the L_1 norm of the *gain vector*, i.e., $\boldsymbol{p}_k - \lambda \boldsymbol{p}_{k-1}$, given by

$$\|\boldsymbol{p}_k - \lambda \boldsymbol{p}_{k-1}\|_1 = \|\delta_k \boldsymbol{x}_k d_k\|_1 \tag{8}$$

undergoes a sudden increase when d_k is corrupted by impulsive noise. As a result, the L_1 norm of $\boldsymbol{p}_k$ is also increased which would, in turn, increase the L_1 norm of $\boldsymbol{w}_k$ in (4). The effect of impulsive noise on (3) caused by d_k can be suppressed by imposing a time-varying upper bound γ_k on the L_1 norm of the gain vector in (8). In other words, we choose δ_k such that the update of the crosscorrelation vector in (3) satisfies the condition

$$\|\boldsymbol{p}_k - \lambda \boldsymbol{p}_{k-1}\|_1 \le \gamma_k. \tag{9}$$

Parameter γ_k is chosen as

$$\gamma_k = \left| \frac{d_k}{e_k} \right| \tag{10}$$

for all k on the basis of extensive simulations. The condition in (9) is satisfied if δ_k is chosen as

$$\delta_k = \frac{1}{\|\boldsymbol{x}_k e_k\|_1}. \tag{11}$$

As can be seen, δ_k can be greater than unity which would affect the convergence performance of the adaptive filter. To circumvent this problem, we use

$$\delta_k = \min\left(1, \frac{1}{\|\boldsymbol{x}_k e_k\|_1}\right). \tag{12}$$

With $\delta_k = 1$, the update equations in (5) and (6) become identical with those of the conventional RLS adaptation algorithm. The value of δ_k given by (12) will also bound the L_1 norm of the *gain matrix*, i.e., $\boldsymbol{R}_k - \lambda \boldsymbol{R}_{k-1}$, given by

$$\begin{aligned} \|\boldsymbol{R}_k - \lambda \boldsymbol{R}_{k-1}\|_1 &= \left\|\delta_k \boldsymbol{x}_k \boldsymbol{x}_k^T\right\|_1 \\ &= \min\left(\|\boldsymbol{x}_k\|_\infty \|\boldsymbol{x}_k\|_1, \frac{\|\boldsymbol{x}_k\|_\infty}{|e_k|}\right). \end{aligned} \tag{13}$$

As can be seen, for an impulsive-noise corrupted e_k, the L_1 norm of the *gain matrix* would be significantly reduced. Since the probability that $\delta_k = 1$ during the transient state is high and the convergence of the RLS algorithm is fast, the initial convergence of the proposed robust RLS algorithm would also be fast. In addition, the proposed robust RLS algorithm would work with $\delta_k = 1$ during steady state as the amplitude of the error signal, e_k, becomes quite low during steady state. Consequently, the steady-state misalignment of the proposed robust RLS algorithm would be similar to those of conventional RLS adaptation algorithms. However, when an impulsive noise-corrupted e_k occurs, we obtain $|d_k| \approx |e_k|$ and $\delta_k = 1/\|\boldsymbol{x}_k e_k\|_1$ which would force the L_1 norm of the gain vector in (8) and the L_1 norm of the *gain matrix* in (13) to be bounded by $\gamma_k \approx 1$ and $\|\boldsymbol{x}_k\|_\infty / |e_k|$, respectively. As a result, the L_1 norm of the coefficient vector $\boldsymbol{w}_k$ in (4) would also remain bounded as discussed below.

The L_1 norm of the differential-coefficient vector of the conventional RLS algorithm given by

$$\triangle \boldsymbol{w}_k = \boldsymbol{w}_k - \boldsymbol{w}_{k-1} \tag{14}$$

is obtained as

$$\|\triangle \boldsymbol{w}_k\|_1 = \frac{|e_k| \|\boldsymbol{S}_{k-1}\boldsymbol{x}_k\|_1}{\left\|\lambda + \boldsymbol{x}_k \boldsymbol{S}_{k-1} \boldsymbol{x}_k^T\right\|_1} \tag{15}$$

by using (6) in (14) with $\delta_k = 1$. As can be seen, the L_1 norm of the differential-coefficient vector in the conventional RLS algorithm increases abruptly for an impulsive noise corrupted e_k. Similarly, the L_1 norm of the differential-coefficient vector in the proposed robust RLS algorithm for the case of an impulsive noise corrupted error signal, e_k, is obtained by using (11) and (6) in (14) as

$$\|\triangle \boldsymbol{w}_k\|_1 = \frac{\|\boldsymbol{S}_{k-1}\boldsymbol{x}_k\|_1}{\|\boldsymbol{x}_k\|_1 \left\|\lambda + \delta_k \boldsymbol{x}_k \boldsymbol{S}_{k-1} \boldsymbol{x}_k^T\right\|_1}. \tag{16}$$

As can be seen, the L_1 norm given by (16) would be much less than that in (15) since e_k cannot perturb $\boldsymbol{S}_{k-1}$. Although δ_k would become less than one in such a situation, its effect is significantly reduced by $\|\boldsymbol{x}_k\|_1$ in (16). It should also be noted that the duration of e_k would have no effect on (16). In other words, the proposed robust RLS algorithm would exhibit robust performance with respect to a long burst of impulsive noise. Using the well known vector-norm inequality

$$\frac{1}{\sqrt{M}} \|\triangle \boldsymbol{w}_k\|_1 \le \|\triangle \boldsymbol{w}_k\|_2 \le \|\triangle \boldsymbol{w}_k\|_1 \tag{17}$$

and (16), we note that the L_2 norm of the differential-coefficient vector would also remain bounded and hence the L_2 norm of $\boldsymbol{w}_k$ in the proposed RLS algorithm would also be robust with respect to the amplitude and duration of the impulsive-noise corrupted e_k.

B. Robust RLS Algorithm for Nonstationary Environments

The above RRLS algorithm, like other RLS algorithms cannot track sudden system disturbances as λ_f is chosen to be very close to unity in order to achieve a reduced steady-state misalignment. To overcome this problem, we use time-varying parameters λ_k and δ_k defined as

$$\lambda_k = \max\left[0.1, \min\left(\lambda_f, \frac{\theta_{1,k} \boldsymbol{x}_k^T \boldsymbol{S}_{k-1} \boldsymbol{x}_k}{\theta_{2,k} - \theta_{1,k} + \theta_{1,k} \boldsymbol{x}_k^T \boldsymbol{S}_{k-1} \boldsymbol{x}_k}\right)\right] \tag{18}$$

$$\delta_k = 1 - \lambda_k. \tag{19}$$

In (18), $\theta_{1,k}$ should be greater than $\theta_{2,k}$ in order to render the proposed RLS algorithm applicable to nonstationary environments. Suitable values for $\theta_{1,k}$ and $\theta_{2,k}$ that were found to give good results in practice are $\theta_{1,k} = 2.24\sigma_{1,e}$ and $\theta_{2,k} = \sigma_{2,e}$. Constant 2.24 is an empirical constant which is chosen to ensure that the probability that $\lambda_k \neq \lambda_f$ is of the order of 0.001. This would ensure that under sudden system disturbances, λ_k would be reduced momentarily and then be quickly returned to the value λ_f in order to maintain the tracking of the algorithm.

The variances of the error signal, $\sigma_{1,e}^2$ and $\sigma_{2,e}^2$, in iteration k are estimated as

$$\sigma_{1,k}^2 = \beta\sigma_{1,k-1}^2 + (1-\beta)\min\left[\sigma_{1,k-1}^2, \text{median}(\boldsymbol{g}_k)\right] \tag{20}$$

$$\sigma_{2,k}^2 = \varsigma\sigma_{2,k-1}^2 + (1-\varsigma)\text{median}(\boldsymbol{g}_k) \tag{21}$$

where $\boldsymbol{g}_k^T = [e_k^2 + \epsilon, \ldots, e_{k-P+1}^2 + \epsilon]$ is a vector of dimension P, $\epsilon \approx 0$ is a very small positive scalar, $0 < \beta < 1$, and $0 < \varsigma < 1$; constants β and ς are referred to as *memory factors* in the literature. In the proposed algorithm, we use λ_k and δ_k given in (18) and (19), respectively, only when $\sqrt{\min(\boldsymbol{g}_k)} > 4\sigma_{1,k}$. Otherwise, we use λ_f and δ_k as given in (12).

If $\sigma_{1,0}^2$ is chosen to be very large, then we would get $\sqrt{\min(\boldsymbol{g}_k)} < 4\sigma_{1,k}$ during the transient state and, therefore, the algorithm would work with λ_f and δ_k as given in (12). As a result, the transient state would die out quickly in which case $\sigma_k^2 \approx \sigma_v^2$ at steady state. On the other hand, for sudden system disturbances, we would get $\sqrt{\min(\boldsymbol{g}_k)} \gg 4\sigma_{1,k}$ in which case the algorithm would work with λ_k and δ_k given in (18) and (19), respectively. In such a situation, λ_k momentarily becomes significantly less than λ_f as $\theta_{2,k} \gg \theta_{1,k}$ and shortly afterwards λ_k becomes equal to λ_f as $\theta_{2,k}$ becomes less than $\theta_{1,k}$. As a result, improved tracking is achieved in nonstationary environments.

C. Discussion

The two versions of the proposed algorithm essentially solve the minimization problem

$$\underset{\boldsymbol{w}_k}{\text{minimize}}\ J_{\boldsymbol{w}_k} = \sum_{i=1}^{k}\delta_i \prod_{j=i+1}^{k}\lambda_j\left(d_i - \boldsymbol{w}_k^T\boldsymbol{x}_i\right)^2 + \frac{1}{2}\left(\prod_{i=1}^{k}\lambda_i\right)\boldsymbol{w}_k^T\boldsymbol{R}_0\boldsymbol{w}_k \tag{22}$$

and they can be implemented in terms of the algorithm summarized in Table I.

For stationary environments, the proposed algorithm entails $3M^2+4M+5$ multiplications and $2M^2+2M+2$ additions per iteration where M is the dimension of the coefficient vector. On the other hand, for nonstationary environments, $3M^2+4M+13$ multiplications and $2M^2+2M+5$ additions per iteration are required. The conventional RLS algorithm requires $3M^2+4M+2$ multiplications and $2M^2+2M$ additions whereas the RLM algorithm requires $3M^2+4M+11$ multiplications and $2M^2+2M+2$ additions. Evidently, for values of M in excess of 5, the computational complexity of the proposed robust RLS algorithm is similar to that of the RLS and RLM algorithms.

TABLE I
IMPLEMENTATION OF PROPOSED ROBUST RLS ALGORITHM

Given d_k and $\boldsymbol{x}_k$ choose P, λ_f, $\boldsymbol{S}_0 = \epsilon^{-1}\boldsymbol{I}$, and compute
$e_k = d_k - \boldsymbol{w}_{k-1}^T\boldsymbol{x}_k$
$\boldsymbol{t}_k = \boldsymbol{S}_{k-1}\boldsymbol{x}_k$
$\tau_k = \boldsymbol{x}_k^T\boldsymbol{t}_k$
$\tilde{\tau}_k = \frac{\lambda_k}{\delta_k} + \tau_k$
$\tilde{\boldsymbol{t}}_k = \frac{1}{\tilde{\tau}_k}\boldsymbol{t}_k$
$\boldsymbol{S}_k = \frac{1}{\lambda_k}\left(\boldsymbol{S}_{k-1} - \tilde{\boldsymbol{t}}_k\boldsymbol{t}_k^T\right)$
$\boldsymbol{w}_k = \boldsymbol{w}_{k-1} + e_k\tilde{\boldsymbol{t}}_k$
For applications in stationary environments, compute
$\lambda_k = \lambda_f$
$\Vert\boldsymbol{x}_k\Vert_1 = \Vert\boldsymbol{x}_{k-1}\Vert_1 + \vert x_k\vert - \vert x_{k-M}\vert$
$\delta_k = \min\left(1, \frac{1}{\Vert\boldsymbol{x}_k\Vert_1\vert e_k\vert}\right)$
For applications in nonstationary environments, compute
$\boldsymbol{g}_k = \left[e_k^2 + \epsilon,\ \ \boldsymbol{g}_{k-1}^T(1, 1:P-1)\right]^T$
$\mathcal{C} = \text{median}(\boldsymbol{g}_k)$
$\sigma_{1,k}^2 = \beta\sigma_{1,k-1}^2 + (1-\beta)\min(\sigma_{1,k-1}^2, \mathcal{C})$
$\sigma_{2,k}^2 = \varsigma\sigma_{2,k-1}^2 + (1-\varsigma)\mathcal{C}$
if $\sqrt{\min(\boldsymbol{g}_k)} > 4\sigma_{1,k}$ let
$\lambda_k = \max\left[0.1, \min\left(\lambda_f, \frac{\theta_{1,k}\tau_k}{\theta_{2,k} - \theta_{1,k} + \theta_{1,k}\tau_k}\right)\right]$
$\delta_k = 1 - \lambda_k$
else let
$\lambda_k = \lambda_f$
$\delta_k = \min\left(1, \frac{1}{\Vert\boldsymbol{x}_k\Vert_1\vert e_k\vert}\right)$
end

III. SIMULATION RESULTS

In this section, the proposed robust RLS (PRRLS) algorithm is compared with the conventional RLS algorithm and the RLM algorithm [4] in terms of robustness and tracking in a system identification application in stationary and nonstationary environments. The unknown system was an FIR filter.

The first experiment concerned the case of a stationary environment. The coefficient vector of the unknown system, $\boldsymbol{w}_{\text{opt}}$, was obtained using MATLAB commands $\boldsymbol{h} = \text{fir1}(M-1, \omega_n)$ and $\boldsymbol{w}_{\text{opt}} = \boldsymbol{h}/\text{norm}(\boldsymbol{h}, 2)$ with $M = 37$ and $\omega_n = 0.3$. The input signal was a zero-mean white Gaussian noise signal with unity variance and was colored by an IIR filter with a single pole at 0.95. The measurement noise added to the desired signal was a zero-mean white Gaussian noise signal with variances $\sigma_v^2 = 10^{-3}$ and 10^{-6} to achieve signal-to-noise ratios (SNRs) of 30 and 60 dB, respectively. The impulsive noise was generated as $\eta_k = \omega_k\nu_k$ where ω_k is a Bernoulli process with the probability that $\omega_k = 1$ is equal to $p = 0.01$, i.e., $P(\omega_k = 1) = 0.01$, and ν_k is a zero-mean Gaussian signal with variance $\sigma_\nu^2 = 10\,000\sigma_y^2$ where σ_y^2 is the power of the uncorrupted output signal [6]. The learning curves obtained in 1000 independent trials by using the conventional RLS and RLM algorithms and the PRRLS algorithm are illustrated in Fig. 1(a)–(b). As can be seen, the RLS and RLM algorithms are not robust with respect to a long burst of impulsive noise whereas the PRRLS algorithm is not affected by the impulsive noise.

The second experiment concerned the case of a nonstationary environment. The initial algorithm parameters were the same as those in the first experiment except that the order of the unknown system was increased to 63 and the variances of the measurement noise were changed to 10^{-4} and 10^{-7} to achieve SNRs of 40 and 70 dB, respectively. The impulse response of the FIR filter was suddenly multiplied by -1 at iteration 1500. The tracking of the robust algorithms in [3]–[6] was examined using a similar setting. The learning curves obtained in 1000 independent trials by using the conventional RLS and RLM algorithms and the PRRLS algorithm are illustrated in Fig. 2(a)–(b). As can be seen, the RLS and RLM algorithms cannot track sudden system changes whereas the PRRLS algorithm handles sudden

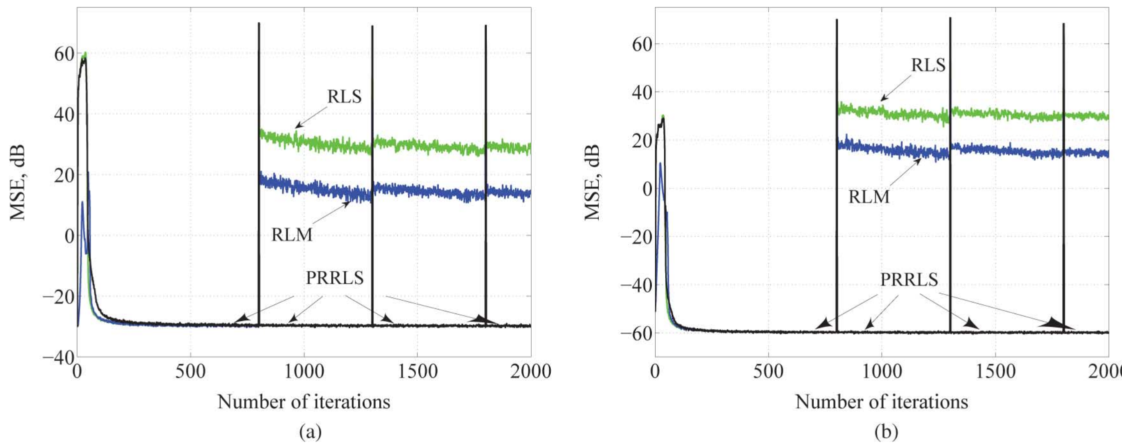

Fig. 1. Learning curves with $\lambda_f = 0.999999$, $\boldsymbol{S}_0 = \epsilon^{-1}\boldsymbol{I}$ with $\epsilon = 10^{-12}$, $p = 0.01$, and $\boldsymbol{w}_0 = \boldsymbol{0}$ in all algorithms. The parameters for the RLM algorithm were $L = 5$, $\lambda_\sigma = 0.95$, $\xi = 1.960\sigma_e$, $\triangle_1 = 2.240\sigma_e$, $\triangle_2 = 2.576\sigma_e$ as suggested in [4]. Impulsive noise of duration $3T_S$ was added to the desired signal at iterations 800, 1400, 1800 where T_s is the sampling period. (a) $\mathrm{SNR} = 30$ dB, (b) $\mathrm{SNR} = 60$ dB.

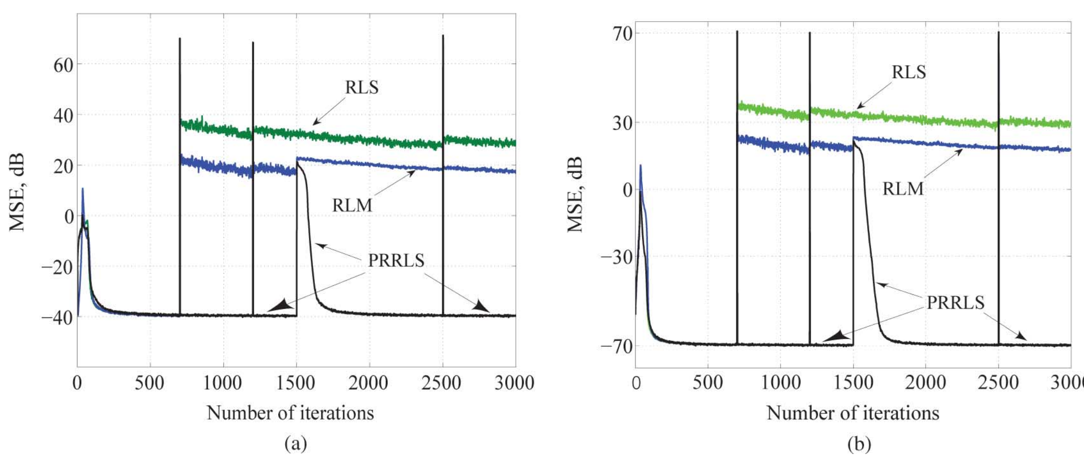

Fig. 2. Learning curves with $P = 5$, $\sigma_{1,0} = \sigma_{2,0} = 1000$, $\beta = 0.99$, $\varsigma = 0.95$, for the PRRLS algorithm. Impulsive noise of duration $3T_S$ was added to the desired signal at iterations 700, 1200, 2500. The initial algorithm parameters were set to $\lambda_f = 0.999999$, $\boldsymbol{S}_0 = \epsilon^{-1}\boldsymbol{I}$ with $\epsilon = 10^{-4}$, and $\boldsymbol{w}_0 = \boldsymbol{0}$ in all algorithms. (a) $\mathrm{SNR} = 40$ dB, (b) $\mathrm{SNR} = 70$ dB.

system changes successfully and at the same time maintains its robustness with respect to impulsive noise.

IV. Conclusion

A new robust RLS adaptive-filtering algorithm that performs well in impulsive noise environments has been proposed. The new algorithm uses the L_1 norm of the gain factor of the cross-correlation vector to achieve robust performance against impulsive noise. In addition, the proposed algorithm uses a modified variance estimator to compute a threshold that is used to obtain a variable forgetting factor λ_k which offers improved tracking. Simulation results show that the proposed algorithm is robust against impulsive noise and offers better tracking compared to the conventional RLS and RLM algorithms.

Acknowledgment

The authors are grateful to the Natural Sciences and Engineering Research Council of Canada for supporting this work.

References

[1] P. S. R. Diniz, *Adaptive Filtering: Algorithms and Practical Implementation*. New York: Springer, 2008.
[2] A. H. Sayed, *Fundamentals of Adaptive Filtering*. New York: Wiley, 2003.
[3] Y. Zou and S. C. Chan, “A robust quasi-Newtion adaptive filtering algorithm for impulse noise suppression,” in *IEEE Int. Symp. Circuits and Syst.*, May 2001, pp. 677–680.
[4] 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 Process. Lett.*, vol. 7, no. 11, pp. 324–326, Nov. 2000.
[5] S. C. Chan and Y. Zou, “A recursive least M-estimate algorithm for robust adaptive filtering in impulsive noise: Fast algorithm and convergene analysis,” *IEEE Trans. Signal Process.*, vol. 52, no. 4, pp. 975–991, Apr. 2004.
[6] L. R. Vega, H. Rey, J. Benesty, and S. Tressens, “A fast robust recurssive least-squares algorithm,” *IEEE Trans. Signal Process.*, vol. 57, no. 3, pp. 1209–1215, Mar. 2009.
[7] P. J. Rousseeuw and A. M. Leroy, *Robust Regression and Outlier Detection*. New York: Wiley, 1987.