IMPROVED CONVERGENCE PERFORMANCE OF ADAPTIVE ALGORITHMS THROUGH LOGARITHMIC COST

Muhammed O. Sayin, N. Denizcan Vanli, Suleyman S. Kozat

Bilkent University, Ankara, Turkey

ABSTRACT

We present a novel family of adaptive filtering algorithms based on a relative logarithmic cost. The new family intrinsically combines the higher and lower order measures of the error into a single continuous update based on the error amount. We introduce the least mean logarithmic square (LMLS) algorithm that achieves comparable convergence performance with the least mean fourth (LMF) algorithm and overcomes the stability issues of the LMF algorithm. In addition, we introduce the least logarithmic absolute difference (LLAD) algorithm. The LLAD and least mean square (LMS) algorithms demonstrate similar convergence performance in impulse-free noise environments while the LLAD algorithm is robust against impulsive interference and outperforms the sign algorithm (SA).

Index Terms— Logarithmic cost function, robustness against impulsive noise, stable adaptive method

I. INTRODUCTION

Adaptive filtering algorithms define certain statistical measure of the error signal, denoting the difference between the observed input and estimated output, as the cost and minimize the cost iteratively through certain update rules. The conventional least mean square (LMS) algorithm uses the mean square error that has mathematical tractability and relative ease of analysis. Based on the famous LMS algorithm, there are alternative adaptive filtering algorithms which reduce the convergence time, e.g., the least mean fourth (LMF) algorithm; or reduce the computational complexity while providing robustness against outliers, e.g., the sign algorithm (SA) [1].

The LMF algorithm uses the fourth power of the error as the cost function [2] and achieves better trade-off between the transient and steady-state performance, however, has certain stability issues [3]. In [3], authors propose the stable normalised LMF algorithm, which might also be derived through the proposed relative logarithmic error cost framework as shown in this paper.

The performance of the least-squares algorithms degrades severely when the input and desired signal pairs are perturbed by heavy tailed impulsive interferences, e.g., in applications involving high power noise signals. The SA uses the L_1 norm of the error as the cost and is robust against impulsive interferences since its update involves only the sign of e_t. However, the SA usually exhibits slower convergence performance especially for highly correlated input signals [1].

In this paper, we present a new family of adaptive filters proposed in [4]. We use *diminishing return* functions, e.g., the logarithm function, as a normalization (or a regularization) term, i.e., as a subtracting term, in the cost function in order to improve the convergence performances. We particularly choose the logarithm function as the normalizing diminishing return function [5] in our cost definitions since the logarithmic function is differentiable and results efficient and mathematically tractable adaptive algorithms. By using the logarithm function, we are able to use of the higher-order statistics of the error for small perturbations. Furthermore, for larger error values, the introduced algorithms seek to minimize the conventional cost functions due to the decreasing weight of the logarithmic term with the increasing error amount. In this sense, the new framework is akin to a continuous generalization of the switched norm algorithms, hence greatly improve the convergence performance of the mixed-norm methods [6], [7] as shown in this paper.

II. PROBLEM DESCRIPTION

The mixed-norm algorithms minimize a combination of different error norms in order to achieve improved convergence performance [6], [7]. Even though the combination parameter brings in an extra degree of freedom, the design of the mixed norm filters requires the optimization of the mixing parameter based on a priori knowledge of the input and noise statistics. On the other hand, the logarithmic cost intrinsically combines cost with different order of error measures based on the error amount.

The impulsive interferences severely degrade the algorithmic updates. In general, the samples contaminated with impulses contain little useful information. Hence, the robust algorithms need to be less sensitive only against large perturbations on the error and can be as sensitive as the conventional least squares algorithms for small error values. The switched-norm algorithms switch between the L_1 and L_2 norms based on the error amount such as the robust Huber filter [8]. This approach combines the better convergence of L_2 and the robustness of L_1 together in a discrete manner with a breaking point in the cost function. We propose a continuous cost function to avoid possible anomalies that might arise due to the breaking points.

In the next section, we introduce the logarithmic cost framework.

III. COST FUNCTION WITH LOGARITHMIC ERROR

Consider the system identification framework where we observe an unknown vector[1] $\mathbf{w}_o \in \mathbb{R}^p$ through a linear model

$$d_t = \mathbf{w}_o^T \mathbf{x}_t + n_t,$$

where n_t represents the noise and $\mathbf{x}_t \in \mathbb{R}^p$ is the regression signal. In the logarithmic cost framework, we estimate the unknown system vector $\mathbf{w}_o$ through the minimization of the following cost function:

$$J\left(e_t\right) \triangleq F\left(e_t\right) - \frac{1}{\alpha} \ln\left(1 + \alpha F\left(e_t\right)\right), \qquad (1)$$

where $e_t \triangleq d_t - \hat{d}_t$ denotes the error between the desired signal d_t and the estimation $\hat{d}_t$, $\alpha > 0$ is a design parameter, and $F\left(e_t\right)$ is a conventional cost function, e.g., $F(e_t) = E\left[|e_t|\right]$ or $F(e_t) = E\left[e_t^2\right]$.

In [9], the authors propose a stochastic cost function using the logarithm function as follows

$$J_{[9]}(e_t) \triangleq \frac{1}{2\gamma} \ln\left(1 + \gamma \left(\frac{e_t}{\|\mathbf{x}_t\|}\right)^2\right)$$

where $\gamma > 0$ is a design parameter. Note that the cost function $J_{[9]}(e_t)$ is the subtracted term in (1) for $F(e_t) = \frac{e_t^2}{\|\mathbf{x}_t\|^2}$. The Hessian matrix of $J_{[9]}(e_t)$ is given by

$$\mathbf{H}\left(J_{[9]}(e_t)\right) = \frac{\mathbf{x}_t \mathbf{x}_t^T}{\|\mathbf{x}_t\|^2 \left(1 + \gamma \left(\frac{e_t}{\|\mathbf{x}_t\|}\right)^2\right)}$$

$$\times \left(1 - \frac{2\gamma e_t^2}{\|\mathbf{x}_t\|^2 \left(1 + \gamma \left(\frac{e_t}{\|\mathbf{x}_t\|}\right)^2\right)}\right).$$

We emphasize that $\mathbf{H}\left(J_{[9]}(e_t)\right)$ is positive semi-definite provided that $\gamma \left(\frac{e_t}{\|\mathbf{x}_t\|}\right)^2 \leq 1$, thus, the parameter γ should be chosen carefully to be able to efficiently use the gradient descent algorithms. On the other hand, the Hessian matrix of $J(e_t)$ is given by

$$\mathbf{H}\left(J(e_t)\right) = \mathbf{H}\left(F(e_t)\right) \frac{\alpha F(e_t)}{1 + \alpha F(e_t)} + \frac{\alpha \nabla_{\mathbf{w}} F(e_t) \nabla_{\mathbf{w}} F(e_t)^T}{\left(1 + \alpha F(e_t)\right)^2},$$

which is positive semi-definite provided that $\mathbf{H}\left(F(e_t)\right)$ is a positive semi-definite matrix, which enables the use of the diminishing return property [5] of the logarithm function

---

[1]As a notation, bold lower (or upper) case letters denote the vectors (or matrices). For a vector $\mathbf{a}$ (or matrix $\mathbf{A}$), $\mathbf{a}^T$ (or $\mathbf{A}^T$) is its ordinary transpose. $\|\cdot\|$ and $\|\cdot\|_{\mathbf{A}}$ denote the L_2 norm and the weighted L_2 norm with the matrix $\mathbf{A}$, respectively (provided that $\mathbf{A}$ is positive-definite). $|\cdot|$ is the absolute value operator. We work with real data for notational simplicity. For a random variable x (or vector $\mathbf{x}$), $E[x]$ (or $E[\mathbf{x}]$) represents its expectation. Here, $\mathrm{Tr}(\mathbf{A})$ denotes the trace of the matrix $\mathbf{A}$ and $\nabla_{\mathbf{x}} f(\mathbf{x})$ is the gradient operator.

for stable and robust updates.

Remark 3.1: By Maclaurin series of the natural logarithm for $\alpha F(e_t) \leq 1$, (1) yields

$$J(e_t) = F(e_t) - \frac{1}{\alpha} \left(\alpha F(e_t) - \frac{\alpha^2}{2} F^2(e_t) + \cdots\right)$$

$$= \frac{\alpha}{2} F^2(e_t) - \frac{\alpha^2}{3} F^3(e_t) + \cdots, \qquad (2)$$

which is an infinite combination of the conventional cost function for small values of $F(e_t)$. We emphasize that the cost function (2) yields to the second power of the cost function $F(e_t)$ for small values of the error while for large error values, the cost function $J(e_t)$ resembles $F(e_t)$ as follows:

$$F(e_t) - \frac{1}{\alpha} \ln\left(1 + \alpha F(e_t)\right) \rightarrow F(e_t) \text{ as } e_t \rightarrow \infty,$$

since the cost $F(e_t)$ increases with increasing error amount. Hence, the new methods are the combinations of the algorithms with mainly $F^2(e_t)$ or $F(e_t)$ cost functions based on the error amount. It is important to note that the objective functions $F^2(e_t)$, e.g., $E[e_t^2]^2$, and $F(e_t^2)$, e.g., $E[e_t^4]$, yields the same stochastic gradient update after removing the expectation in this paper.

IV. NEW ALGORITHMS

Based on the gradient of $J(e_t)$ we obtain the general steepest descent update as

$$\mathbf{w}_{t+1} = \mathbf{w}_t - \mu \nabla_{\mathbf{w}} F(e_t) \frac{\alpha F(e_t)}{1 + \alpha F(e_t)},$$

where $\mu > 0$ is the step size and α is a positive design parameter with a typical value $\alpha = 1$. If we assume that after removing the expectation to generate stochastic gradient updates $F(e_t)$ yields $f(e_t)$, e.g., $F(e_t) = E[f(e_t)]$, then the general stochastic gradient update is given by

$$\mathbf{w}_{t+1} = \mathbf{w}_t + \mu \mathbf{x}_t \nabla_{e_t} f(e_t) \frac{\alpha f(e_t)}{1 + \alpha f(e_t)}. \qquad (3)$$

In the following subsections, we introduce algorithms improving the performance of the conventional algorithms such as the LMS (i.e. $f(e_t) = e_t^2$), sign algorithm (i.e. $f(e_t) = |e_t|$) and normalized updates.

IV-A. The Least Mean Logarithmic Square (LMLS) Algorithm

For $F(e_t) = E[e_t^2]$, the stochastic gradient update yields

$$\mathbf{w}_{t+1} = \mathbf{w}_t + \mu \frac{\alpha \mathbf{x}_t e_t^3}{1 + \alpha e_t^2}. \qquad (4)$$

Note that we include the multiplier '2' coming from the gradient $\nabla_{e_t} e_t^2 = 2e_t$ into the step-size μ. The algorithm (4) resembles a least-mean fourth update for the small error values while it behaves like the least-mean square algorithm for large perturbations on the error.

IV-B. The Least Logarithmic Absolute Difference (LLAD) Algorithm

The SA utilizes $F(e_t) = E[|e_t|]$ as the cost function, which provides robustness against impulsive interferences [10]. However, the SA has slower convergence rate since the L_1 norm is the smallest possible error power for a convex cost function. In the logarithmic cost framework, for $F(e_t) = E[|e_t|]$, (3) yields

$$\mathbf{w}_{t+1} = \mathbf{w}_t + \mu \frac{\alpha \mathbf{x}_t e_t}{1 + \alpha |e_t|}. \tag{5}$$

The algorithm (5) combines the LMS algorithm and SA into a single robust algorithm with improved convergence performance.

IV-C. Normalized Updates

For $F(\frac{e_t}{\|\mathbf{x}_t\|}) = E\left[(\frac{e_t}{\|\mathbf{x}_t\|})^2\right]$, we get the normalized least mean logarithmic square (NLMLS) algorithm given by

$$\mathbf{w}_{t+1} = \mathbf{w}_t + \frac{\mu \alpha \mathbf{x}_t e_t^3}{\|\mathbf{x}_t\|^2 \left(\|\mathbf{x}_t\|^2 + \alpha e_t^2\right)}. \tag{6}$$

We point out that (6) is also proposed as the stable normalized least mean-fourth algorithm in [3].

For $F(\frac{e_t}{\|\mathbf{x}_t\|}) = E\left[\frac{|e_t|}{\|\mathbf{x}_t\|}\right]$, we obtain the normalized least logarithmic absolute difference (NLLAD) algorithm as

$$\mathbf{w}_{t+1} = \mathbf{w}_t + \frac{\mu \alpha \mathbf{x}_t e_t}{\|\mathbf{x}_t\| \left(\|\mathbf{x}_t\| + |e_t|\right)}. $$

Next, we provide the stability bound for the learning rate of the LMLS algorithm and provide steady-state analysis of the LLAD algorithm in the impulsive noise environments.

IV-D. Stability Bound for the LMLS Algorithm

In [4], we show that the step size bound for the LMLS is given by

$$\mu \leq \frac{1}{E\left[\|\mathbf{x}_t\|^2\right]} \inf_{E[e_{a,t}^2] \in \Omega} \left\{ \frac{E[e_{a,t} e_t]}{E\left[e_t^2\right]} \beta \right\},$$

where $e_{a,t} = \mathbf{x}_t^T (\mathbf{w}_o - \mathbf{w}_t)$ denotes the *a priori* error and

$$\beta \triangleq \frac{E\left[\frac{\alpha e_t^4}{1+\alpha e_t^2}\right]}{E\left[\frac{\alpha^2 e_t^6}{(1+\alpha e_t^2)^2}\right]}$$

$$= \frac{E\left[\frac{\alpha e_t^4}{(1+\alpha e_t^2)^2}\right] + E\left[\frac{\alpha^2 e_t^6}{(1+\alpha e_t^2)^2}\right]}{E\left[\frac{\alpha^2 e_t^6}{(1+\alpha e_t^2)^2}\right]} \geq 1.$$

We emphasize that the LMLS extends the stability bound of the LMS algorithm (the same bound with $\beta = 1$) while performing comparable performance with the LMF algorithm, which has several stability issues [3].

IV-E. Robustness Analysis for the LLAD Algorithm

In order to analyze the performance in the impulsive noise environments, we use the following model.

Impulsive noise model: We model the noise as a summation of two independent random terms [11] as

$$n_t = n_{o,t} + b_t n_{i,t},$$

where $n_{o,t}$ is the ordinary noise signal that is zero-mean Gaussian with variance $\sigma_{n_o}^2$ and $n_{i,t}$ is the impulse-noise that is also zero-mean Gaussian with significantly large variance $\sigma_{n_i}^2$. Here, b_t is generated through a Bernoulli random process and determines the occurrence of the impulses in the noise signal with $p_B(b_t = 1) = \nu_i$ and $p_B(b_t = 0) = 1 - \nu_i$ where ν_i is the frequency of the impulses in the noise signal. The corresponding probability density function is given by

$$p_n(n_t) = \frac{1 - \nu_i}{\sqrt{2\pi}\sigma_{n_o}} \exp\left(-\frac{n_t^2}{2\sigma_{n_o}^2}\right) + \frac{\nu_i}{\sqrt{2\pi}\sigma_n} \exp\left(-\frac{n_t^2}{2\sigma_n^2}\right),$$

where $\sigma_n^2 = \sigma_{n_o}^2 + \sigma_{n_i}^2$.

We particularly analyze the steady-state performance of the LLAD algorithm (for which $f(e_t) = |e_t|$) in the impulsive noise environments since we motivate the LLAD algorithm as improving the steady state convergence performance of the SA. Based on the impulsive noise model, in [4], we provide the steady-state excess mean square error (EMSE) of the LLAD algorithm as follows

$$\zeta_{\text{LLAD}}^* = \frac{\mu \text{Tr}(\mathbf{R}) \left(\nu_i + \alpha^2(1 - \nu_i)\sigma_{n_o}^2\right)}{\alpha(1 - \nu_i)(2 - \alpha\mu\text{Tr}(\mathbf{R})) + \sqrt{\frac{8}{\pi}} \frac{\nu_i}{\sigma_n}}. \tag{7}$$

**Remark 4.1:** Increasing ν_i or in other words more frequent impulses cause larger steady state EMSE. However, through the optimization of α, we can minimize the steady state EMSE (7). After some algebra, the optimum design parameter in impulsive noise environment is roughly given by

$$\alpha_{\text{opt}} \approx \sqrt{\frac{\nu_i}{1 - \nu_i}} \frac{1}{\sigma_{n_o}}.$$

V. NUMERICAL EXAMPLES

We particularly compare the convergence rate of the algorithms for the same steady state MSD through the specific choice of the step sizes for a fair comparison. Here, we have a stationary data $d_t = \mathbf{w}_o^T \mathbf{x}_t + n_t$ where $\mathbf{x}_t$ is zero-mean Gaussian i.i.d. regression signal with variance $\sigma_x^2 = 1$, n_t represents zero-mean Gaussian i.i.d. noise signal with the variance $\sigma_n^2 = 0.01$ and the parameter of interest $\mathbf{w}_o \in \mathbb{R}^5$ is randomly chosen.

In Fig. 1, we compare the convergence rate of the LMLS, LMF and LMS algorithms for small and relatively large step sizes. Fig.1a shows that LMLS and LMF algorithms achieve comparable performance and LMLS achieves better convergence performance than the LMS algorithm. In Fig. 1b, we compare the LMLS and LMS algorithms for relatively large step sizes. We only include the LMLS and LMS algorithms
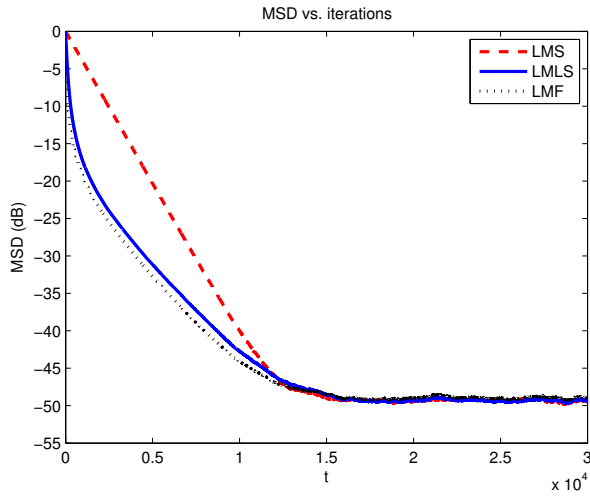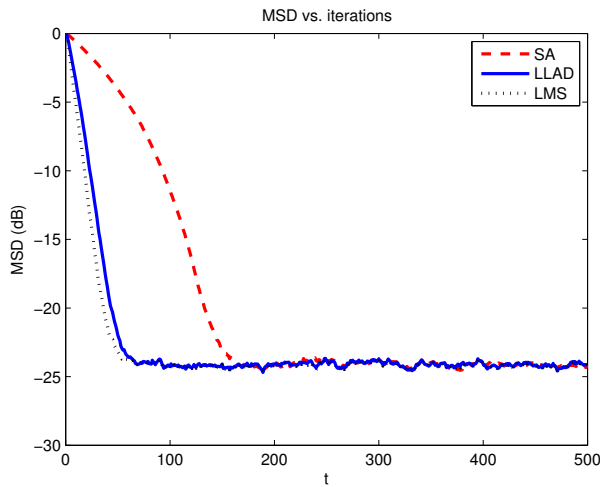
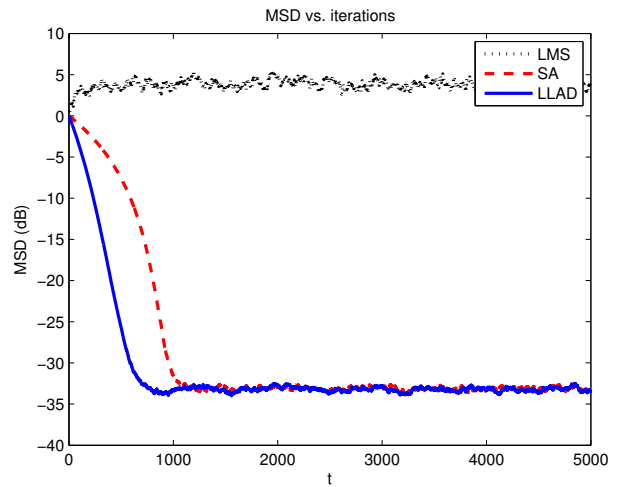(a) $\mu_{\text{LMLS}} = \mu_{\text{LMF}} = 0.01$ and $\mu_{\text{LMS}} = 0.00047$.
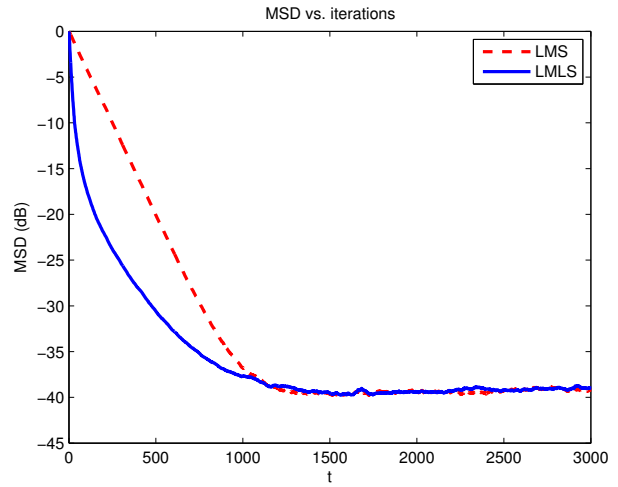
(b) $\mu_{\text{LMLS}} = 0.1$ and $\mu_{\text{LMS}} = 0.0047$ (LMF is unstable for $\mu_{\text{LMF}} = 0.1$).

Fig. 1. Comparison of the MSD of the LMLS, LMS and LMF algorithms.

(a) Impulse-free noise environment, $\alpha = 1$.

(b) 1% impulsive noise environment, $\alpha_{opt} = 1.005$

Fig. 2. Comparison of the MSD of the LLAD, SA and LMS algorithms.

since the LMF algorithm is not stable for such a step-size. Hence, the LMLS algorithm demonstrates comparable convergence performance with the LMF algorithm with extended stability bound.

In Fig. 2, we compare the LLAD, SA and LMS algorithms in impulse-free and 1% impulsive noise environments. Fig.2a shows that the LLAD algorithm shows comparable convergence performance with the LMS algorithm. In Fig.2b, we use the impulsive noise model with $\sigma_{n_i}^2 = 10^4$ and we observe that the LMS algorithm does not converge while the LLAD algorithm, which achieves comparable convergence performance with the LMS algorithm in the impulse free environment, performs still better than the SA.

VI. CONCLUSION

In this paper, we present a novel family of adaptive filtering algorithms based on the logarithmic error cost framework. We propose important members of the new family, i.e., the LMLS and LLAD algorithms. The LMLS algorithm achieves comparable convergence performance with the LMF algorithm with far larger stability bound on the step size. In the impulse-free environment, the LLAD algorithm has a similar convergence performance with the LMS algorithm. Furthermore, the LLAD algorithm is robust against impulsive interferences and outperforms the SA. Finally, we show the improved convergence performance of the new algorithms in several different system identification scenarios.

VII. REFERENCES

[1] A. H. Sayed, *Fundamentals of Adaptive Filtering*, John Wiley and Sons, 2003.

[2] E. Walach and B. Widrow, "The least mean fourth (LMF) adaptive algorithm and its family," *IEEE Trans. Inform. Theory*, vol. 30, no. 2, pp. 275–283, 1984.

[3] E. Eweda and N. J. Bershad, "Stochastic analysis of a stable normalized least mean fourth algorithm for adaptive noise canceling with a white Gaussian reference," *IEEE Trans. Signal Process.*, vol. 60, no. 12, pp. 6235–6244, 2012.

[4] M. O. Sayin, N. D. Vanli, and S. S. Kozat, "A novel family of adaptive filtering algorithms based on the logarithmic cost," *Available as Arxiv preprint arXiv:1311:6809*, 2013.

[5] R. G. Bartle and D. R. Scherbert, *Introduction to Real Analysis*, John Wiley and Sons, 2011.

[6] J. A. Chambers, O. Tanrikulu, and A. G. Constantinides, "Least mean mixed-norm adaptive filtering," *Electron. Lett.*, vol. 30, no. 19, pp. 1574–1575, 1994.

[7] J. Chambers and A. Avlonitis, "A robust mixed-norm adaptive filter algorithm," *IEEE Signal Process. Lett.*, vol. 4, no. 2, pp. 46–48, 1997.

[8] P. Petrus, "Robust huber adaptive filter," *IEEE Trans. Signal Process.*, vol. 47, no. 4, pp. 1129–1133, 1999.

[9] I. Song, P. Park, and R. W. Newcomb, "A normalized least mean squares algorithm with a step-size scaler against impulsive measurement noise," *IEEE Trans. Circuits Syst. II: Express Briefs*, vol. 60, no. 7, pp. 442–445, 2013.

[10] V. John Mathews and Sung-Ho Cho, "Improved convergence analysis of stochastic gradient adaptive filters using the sign algorithm," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. 35, no. 4, pp. 450–454, 1987.

[11] Xiaodong W. and H. V. Poor, "Joint channel estimation and symbol detection in Rayleigh flat-fading channels with impulsive noise," *IEEE Comm. Lett.*, vol. 1, no. 1, pp. 19–21, 1997.