PERFORMANCE ANALYSIS OF RECURSIVE LEAST MODULI ALGORITHM FOR FAST CONVERGENT AND ROBUST ADAPTIVE FILTERS

Shin'ichi Koike
Consultant, Tokyo, Japan

ABSTRACT

This paper derives a new adaptation algorithm named *recursive least moduli* (RLM) algorithm that combines least mean modulus (LMM) algorithm for complex-domain adaptive filters with recursive estimation of the inverse covariance matrix of the filter reference input. The RLM algorithm achieves significant improvement in the filter convergence speed of the LMM algorithm with a strongly correlated filter reference input, while it preserves robustness of the LMM algorithm against impulsive observation noise. Analysis of the RLM algorithm is developed for calculating transient and steady-state behavior of the filter convergence. Through experiment with simulations and theoretical calculations of the filter convergence for the RLM algorithm, we demonstrate its effectiveness in making adaptive filters fast convergent and robust in the presence of impulse noise. Good agreement between the simulations and theory proves the validity of the analysis.

Index Terms – RLS algorithm, covariance, impulse noise

1. INTRODUCTION

Adaptive filtering technology has been and still is playing an essential role in implementing latest communication systems, e.g., mobile communication systems, digital broadcasting systems, internet access systems, etc.

Among many adaptation algorithms for adaptive filters, the least mean square (LMS) algorithm is most popular and widely applied to practical systems in industry as well as in science [1], [2]. Although the LMS algorithm attracts many implementers because of its superior performance and well-established design practices, one of its weaknesses is known to be vulnerability to disturbances, e.g. impulsive observation noise [3]. For complex-domain adaptive filters, *least mean modulus* (LMM) algorithm is one of the solutions to this problem [4]. However, the LMM algorithm has a serious disadvantage of slower convergence speed.

It is well known that for any adaptation algorithm a strongly correlated filter input makes the filter converge significantly slower because of the wide spread of eigenvalues of input covariance matrix. One of the methods to "de-correlate" the filter input is recursive estimation of the inverse covariance matrix. The recursive least squares (RLS) algorithm effectively accelerates the convergence of LMS adaptive filters, e.g. [2, Chap.8]. However, the RLS algorithm is again vulnerable to impulse noise.

The above observations inspire us to combine the LMM algorithm with the recursive estimation of the inverse covariance matrix, yielding a new algorithm, named *recursive least moduli* (RLM) algorithm, that could realize much faster convergent adaptive filters while it preserves the robustness of the LMM algorithm against impulse noise.

2. ADAPTIVE FILTER PERFORMANCE IN THE PRESENCE OF IMPULSE NOISE

In this section, we first present an impulse noise model and then evaluate adaptive filter performance for the LMS, LMM and RLS algorithms in the presence of impulse noise.

2.1. Impulse Noise Model

Additive impulsive observation noise is often modeled as *contaminated Gaussian noise* (CGN) that is mathematically a combination of two independent Gaussian noise sources [5], i.e.,

$v_0(n)$: Gaussian noise source #0 with variance $\sigma^2 v_0$ and probability of occurrence pv_0, and

$v_1(n)$: Gaussian noise source #1 with variance $\sigma^2 v_1$ and probability of occurrence pv_1.

Note that $pv_0 + pv_1 = 1$. Variance of CGN is given by $\sigma^2 v = pv_0\, \sigma^2 v_0 + pv_1\, \sigma^2 v_1$. Usually, $\sigma^2 v_1 >> \sigma^2 v_0$ and $pv_1 < pv_0$. For "pure" Gaussian noise, $pv_1 = 0$ and $\sigma^2 v = \sigma^2 v_0$.

2.2. LMS and LMM Algorithms

A cost function $L(n) = |\, e(n)\, |^2 / 2$ gives the well-known (complex-domain) LMS algorithm whose tap weight update equation for an FIR-type adaptive filter can be written as

$$\mathbf{c}(n+1) = \mathbf{c}(n) + \alpha_{\mathrm{C}}\, e^*(n)\, \mathbf{a}(n),$$

whereas for a cost function $L(n) = |\, e(n)\, |$ (*modulus* of the error), we derive least mean modulus (LMM) algorithm [4] with tap weight update equation

$$\mathbf{c}(n+1) = \mathbf{c}(n) + \alpha_{\mathrm{C}}\, [e^*(n) \,/\, |\, e(n)\, |]\, \mathbf{a}(n),$$

where n is time instant, $\mathbf{c}(n)$ is tap weight vector (N taps), $\mathbf{a}(n) = [a(n), \cdots, a(n-N+1)]^T$ is filter reference input vector (length N), α_{C} is step size and $(\ \cdot\ )^*$ denotes *complex conjugate*. The error signal is given by $e(n) = \in(n) + v(n)$, where $\in(n) = \boldsymbol{\theta}^H(n)\, \mathbf{a}(n)$ is excess error, $\boldsymbol{\theta}(n) = \mathbf{h} - \mathbf{c}(n)$ is tap weight misalignment vector, $\mathbf{h}$ is impulse response vector of the unknown stationary system (length N) and $v(n)$ is additive observation noise, CGN in general.

As is well known, for a same level of steady-state excess error, the LMS algorithm normally converges faster than the LMM algorithm if the additive noise is "pure" Gaussian noise. In the presence of impulse noise (CGN), the steady-state excess error for the LMM algorithm remains much smaller than that for the LMS algorithm, showing the high robustness of the LMM algorithm [4]. Indeed, with the LMS algorithm the impulse noise is not suppressed at all.

2.3. RLS Algorithm

As stated earlier, the RLS algorithm achieves significant improvement in the convergence speed of the LMS algorithm with a highly correlated filter reference input. We run simulations of the RLS algorithm for the example below.

Example #1 $N = 32$

filter ref. input: AR(1) Gaussian process with variance $\sigma^2 a = 1$ and regression coefficient $\eta = 0.9$

step size: $\alpha c = 1$

forgetting factor: $\lambda = 1 - 2^{-10}$

Case 1: "pure" Gaussian noise $\sigma^2 v = 0.01$

Case 2: CGN $\sigma^2 v_0 = 0.01$; $pv_0 = 0.9$

$\sigma^2 v_1 = 10$; $pv_1 = 0.1$

Results of the simulations for *Example #1* are shown in Fig. 1, where we observe significantly faster convergence than that for the LMS algorithm (step size 2^{-11}). However, in Case 2, the steady-state excess error considerably increases (by 20 dB) as is the case with the LMS algorithm.

If the recursive estimation of the inverse covariance matrix of the filter reference input as used in the RLS algorithm is to be combined with the LMM algorithm, we expect that improvement in the convergence speed could be achieved for strongly correlated filter inputs, while preserving the robustness against the impulsive observation noise.

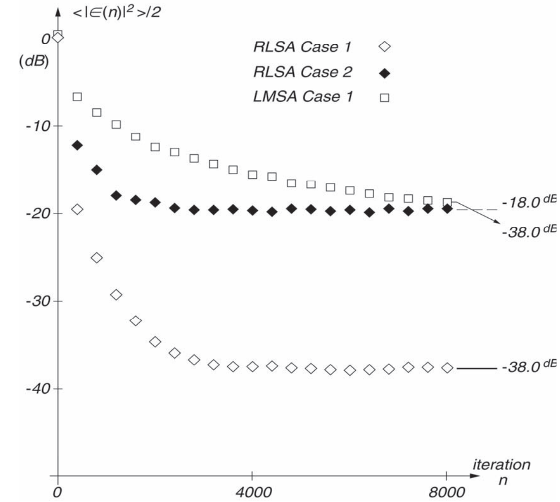

Fig. 1 Adaptive filter convergence. (*Example #1*; RLS, $N = 32$, $\eta = 0.9$ Case 1: "pure" GN, Case 2: CGN)

3. RECURSIVE LEAST MODULI ALGORITHM

Let the cost function of the error be defined as

$$L_F(n) = \Sigma_{i=1}^{n} \lambda^{n-i} F[|e(i)|],$$

where λ is forgetting factor and the function $F(\cdot)$ is monotonically increasing, non-negative, continuous everywhere and differentiable except at a limited number of points. Then we can derive a generalized recursive type algorithm. For the RLS algorithm, we have $F(|e|) = |e|^2/2$.

If $F(|e|) = |e|$, we derive a new adaptation algorithm that is clearly different from the RLS algorithm but is the LMM algorithm combined with the recursive estimation of the inverse covariance matrix. We can derive the tap weight update equation given by

$$\mathbf{c}(n+1) = \mathbf{c}(n) + \alpha c\, [e^*(n) / |e(n)|]\, \mathbf{P}(n)\, \mathbf{a}(n), \quad (1)$$

where we have two kinds of method for updating $\mathbf{P}(n)$ which is an estimate of the inverse covariance matrix.

Method ⟨A⟩ (*Indirect Method*)

$$\mathbf{P}(n+1) = \mathbf{Q}^{-1}(n+1) \quad (2a)$$

with $$\mathbf{Q}(n+1) = \lambda\, \mathbf{Q}(n) + \mathbf{a}(n)\, \mathbf{a}^H(n) / |e(n)| \quad (2b)$$

Method ⟨B⟩ (*Direct Method*)

$$\mathbf{P}(n+1) = \lambda^{-1}\{ \mathbf{P}(n) - \mathbf{P}(n)\, \mathbf{a}(n)\, \mathbf{a}^H(n)\, \mathbf{P}(n) / [\lambda\, |e(n)| + \mathbf{a}^H(n)\, \mathbf{P}(n)\, \mathbf{a}(n)] \} \quad (3)$$

Note that (2b) or (3) contains the error modulus $|e(n)|$ which does not appear in the RLS algorithm.

As the name of the RLS algorithm comes from the "squares" of the error, we name the above algorithm *recursive least moduli* (RLM) algorithm. Though the RLS algorithm has been intensively studied, the proposed RLM algorithm is first dealt with in this paper.

4. ANALYSIS OF RLM ALGORITHM

In this section, we develop performance analysis of adaptive filter convergence for the RLM algorithm. Due to space limitation, detailed derivation process cannot be fully described, but only main results will be summarized. However, the validity of the analytical results in this section will be verified through experiment in Section 5.

4.1. Assumptions

For the analysis to be developed in this section, we make the following assumptions.

A1: The filter reference input $\mathbf{a}(n) = \mathbf{a}_R(n) + j\, \mathbf{a}_I(n)$ is a stationary correlated (or colored) Gaussian process. $\mathbf{a}_R(n)$ and $\mathbf{a}_I(n)$ are independent and identical processes with zero mean, covariance matrix $\mathbf{R}a$ and variance $\sigma^2 a$.

A2: The additive observation noise (CGN in general) is stationary and independent of $\mathbf{a}(n)$.

A3: The filter reference input $\mathbf{a}(n)$ and the tap weights $\mathbf{c}(n)$ are mutually independent (*Independence Assumption*).

A4: The estimate $\mathbf{Q}(n)$, or $\mathbf{P}(n)$, and the filter reference input $\mathbf{a}(n)$ are mutually independent.

A5: The error $e(n)$ and the filter reference input $\mathbf{a}(n)$ are jointly Gaussian distributed [6].

Assumption **A4** is another independence assumption that facilitates analysis of recursive type algorithms.

4.2. Difference Equations for Tap Weights

Under the assumptions above, a set of difference equations for the mean vector $\mathbf{m}(n) = E[\boldsymbol{\theta}(n)]$ and the second-order moment matrix $\mathbf{K}(n) = E[\boldsymbol{\theta}(n)\boldsymbol{\theta}^H(n)]$ of the tap weight misalignment vector $\boldsymbol{\theta}(n)$ can be derived from (1) as follows.

$$\mathbf{m}(n+1) = \mathbf{m}(n) - \alpha_c\, \mathbf{p}(n)$$

$$\mathbf{K}(n+1) = \mathbf{K}(n) - \alpha_c\, [\mathbf{V}(n) + \mathbf{V}^H(n)] + \alpha^2_c\, \mathbf{T}(n),$$

where $\mathbf{p}(n) = E[\mathbf{P}(n)]\mathbf{W}(n)\mathbf{m}(n)$, $\mathbf{K}(n) = E[\mathbf{P}(n)]\mathbf{W}(n)\mathbf{K}(n)$, $\mathbf{T}(n) = 2E[\mathbf{P}(n)]\, \mathbf{R_a}\, E[\mathbf{P}(n)]$, $\mathbf{W}(n) = (\pi/2)^{1/2}\, \sigma^{-1}{}_{e\mathrm{CGN}}(n)\, \mathbf{R_a}$, $\sigma_{e\mathrm{CGN}}(n) = 1 / \Sigma_{i=0}^{\ 1}\, p_{v_i} / \sigma_{e_i}(n)$ and $\sigma^2{}_{e_i}(n) = \varepsilon(n) + \sigma^2{}_{v_i}$ (i = 0, 1). Here, we define excess mean square error (EMSE) as a measure to evaluate adaptive filter performance.

$$\varepsilon(n) = E[|\in(n)|^2] / 2 = \mathrm{tr}[\mathbf{R_a}\, \mathbf{K}(n)],$$

where tr(·) is *trace* of a matrix.

4.3. Calculation of Estimate E[P(n+1)] – Method ⟨A⟩

In this subsection, we assume the absence of impulse noise for simplicity. From (2b) we derive

$$E[\mathbf{Q}(n+1)] = \lambda\, E[\mathbf{Q}(n)] + \mathbf{T_e}(n), \quad (4)$$

where, using Gaussian conditional probability of $e(n)$ given $\mathbf{a}(n)$ and the method used in [7], we calculate

$$\mathbf{T_e}(n) = 2(2/\pi)^{1/2}\, \sigma^{-1}{}_{ec}(n) \int_0^{\pi/2} \mathbf{D_M}(\varphi, n)\, |\mathbf{A_M}(\varphi, n)|^{-1}\, d\varphi$$

with

$$\mathbf{D_M}(\varphi, n) = \mathbf{A_M}^{-1}(\varphi, n)\mathbf{R_a},$$

$$\mathbf{A_M}(\varphi, n) = \mathbf{I} + \mathbf{M_a}(n)\sin^2(\varphi) / \sigma^2{}_{ec}(n),$$

$\mathbf{M_a}(n) = \mathbf{R_a}\, \mathbf{m}(n)\mathbf{m}^H(n)$ and $\sigma^2{}_{ec}(n) = \varepsilon(n) + \sigma^2{}_v - \mathrm{tr}[\mathbf{M_a}(n)]$.

With $E[\mathbf{Q}(n+1)]$, we approximately calculate in (2a)

$$E[\mathbf{P}(n+1)] = E[\mathbf{Q}^{-1}(n+1)] \approx E[\mathbf{Q}(n+1)]^{-1}. \quad (5)$$

This $E[\mathbf{Q}(n+1)]^{-1}$ is known to be less accurate, though Method ⟨A⟩ is used in some papers, e.g. [8].

4.4. Calculation of Estimate E[P(n+1)] – Method ⟨B⟩

We again assume the absence of impulse noise. Under Assumption **A4**, we find from (3) the following difference equation.

$$E[\mathbf{P}(n+1)] = \lambda^{-1}E[\mathbf{P}(n)]\{\mathbf{I} - \boldsymbol{\Phi}(n)E[\mathbf{P}(n)]\}, \quad (6)$$

where

$$\boldsymbol{\Phi}(n) \cong \int_0^{\infty} 2\mathbf{D_{MP}}(\beta, n)\, |\mathbf{A_{MP}}(\beta, n)|^{-1}\, d\beta \cdot \int_0^{\infty} \exp[-\beta\, \lambda\, \sigma_{ec}(n)\, r] \exp(-r^2/2)\, r \cdot I_0\{[(\pi/2)^{1/2} M_{\mathrm{DMP}}{}^{1/2}(\beta, n)/\sigma_{ec}(n)]r\}\, dr \quad (7)$$

with

$$\mathbf{D_{MP}}(\beta, n) = \mathbf{A_{MP}}^{-1}(\beta, n)\mathbf{R_a},$$

$$\mathbf{A_{MP}}(\beta, n) = \mathbf{I} + \mathbf{M_a}(n) / \sigma^2{}_{ec}(n) + 2\,\beta\, \mathbf{R_a}\, E[\mathbf{P}(n)]$$

and $M_{\mathrm{DMP}}(\beta, n) = \mathrm{tr}[\mathbf{D_{MP}}(\beta, n)\mathbf{m}(n)\mathbf{m}^H(n)]$. Here, $I_0(\cdot)$ is the *zero-th order Modified Bessel Function of the first kind* [9].

4.5. Analysis for a Large Number of Tap Weights

For $N >> 1$, we can approximately calculate the expectation for $e(n)$ and $\mathbf{a}(n)$ separately.

For Method ⟨A⟩, we easily calculate

$$\mathbf{T_e}(n) \approx 2\,(\pi/2)^{1/2}\, \sigma^{-1}{}_{e\mathrm{CGN}}(n)\, \mathbf{R_a}. \quad (8)$$

4.6. Initial Conditions

Usually, initial tap weights are selected $\mathbf{c}(0) = \mathbf{0}$. Then, $\mathbf{m}(0) = \mathbf{h}$, $\mathbf{K}(0) = \mathbf{h}\ \mathbf{h}^H$ and $\varepsilon(0) = \mathbf{h}^H\, \mathbf{R_a}\, \mathbf{h}$. We choose initial value of the estimate $\mathbf{P}(0) = P_0\, \mathbf{I}$, where P_0 is determined so that $\varepsilon(1)$ be minimized. We find

$$P_0 = (1/2)(\pi/2)^{1/2}\, \sigma^{-1}{}_{e\mathrm{CGN}}(0)\, \mathbf{h}^H\, \mathbf{R_a}^2\, \mathbf{h} / [\alpha_c\, \mathrm{tr}(\mathbf{R_a}^2)].$$

4.7. Steady-State Solution

Assuming that the impulse noise is absent and the filter converges as $n \to \infty$, we solve steady-state EMSE $\varepsilon(\infty)$.

For Method ⟨A⟩,

$$\varepsilon(\infty) = \delta\, \sigma^2{}_v / (1 - \delta)$$

with

$$\delta = \alpha_c\, \lambda_c\, \pi^{-1} N,$$

where $\lambda_c = 1 - \lambda$ may be called *complementary forgetting factor.*

For Method ⟨B⟩,

$$\delta = \alpha_c\, \lambda_c\, \gamma_P\, \pi^{-1} N,$$

where we find that $\gamma_P > 1$ holds.

5. EXPERIMENT

In this section, experiment is carried out with simulations and theoretical calculations of transient and steady-state behavior of adaptive filter convergence for the RLM algorithm. The effectiveness of the RLM algorithm as well as the validity of the analysis in Section 4 will be demonstrated.

The following two examples are prepared.

Example #2 $N = 4$
- filter ref. input: AR(1) Gaussian process with variance $\sigma^2{}_a = 1$ and regression coefficient $\eta = 0.5$
- step size: $\alpha_c = 1$
- complementary forgetting factor: $\lambda_c = 2^{-7}$
- Case 1: "pure" Gaussian noise $\sigma^2{}_v = 0.01$

Example #3 $N = 32$
- filter ref. input: same as *Example #1*
- step size: $\alpha_c = 1$
- complementary forgetting factor: $\lambda_c = 2^{-10}$
- Case 1 & Case 2: same as *Example #1*

Fig. 2 shows results for *Example #2* (no impulse noise) where the number of tap weights N is small. Theoretical convergence is calculated using (6) and (7) (Method ⟨B⟩). In the figure, theoretical convergence for the LMM algorithm (step size $\alpha_c = 2^{-11}$, $\varepsilon(\infty) = -38.0$ dB) is also plotted for comparison. We observe significantly faster convergence for the RLM algorithm.

In Fig. 3, we depict EMSE convergence for Cases 1 and 2 of *Example #3* (N is large), together with the simulated convergence for the LMM algorithm (step size $\alpha_c = 2^{-14}$, $\varepsilon(\infty) = -38.0$ dB) and for the RLS algorithm (see Case 1 of *Example #1*) where the theoretical convergence is calculated using (4), (5) and (8) (Method ⟨A⟩). The steady-state EMSE for Case 1 is –39.6 dB in theory, whereas that for Case 2 is

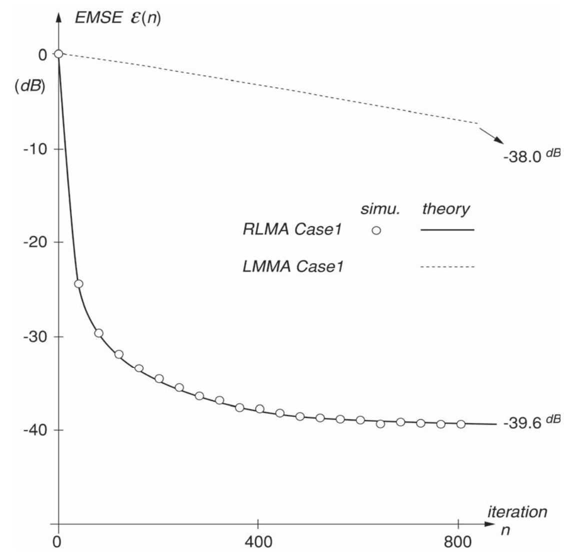

Fig. 2 Adaptive filter convergence.
(*Example #2*; RLM, $N = 4$, $\eta = 0.5$
Case 1: "pure" GN)

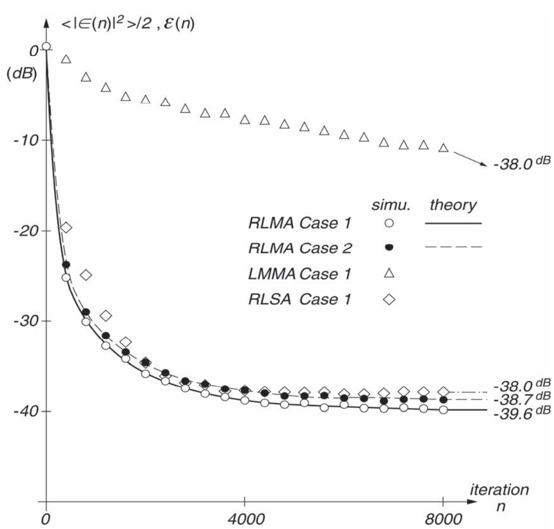

Fig. 3 Adaptive filter convergence.
(*Example #3*; RLM, $N = 32$, $\eta = 0.9$
Case 1: "pure" GN, Case 2: CGN)

–38.7 dB, the increase in $\varepsilon(\infty)$ due to the CGN being within 1 dB. We observe that the filter converges significantly faster than the LMM algorithm, even as fast as the RLS algorithm, while preserving the robustness of the LMM algorithm against impulsive observation noise, that demonstrates the effectiveness of the RLM algorithm.

In Figs. 2 and 3, we see good agreement between the simulated and theoretically calculated convergence of the EMSE that proves the validity of the analysis developed in Section 4 for practical use.

6. CONCLUSION

In this paper, we have derived a new adaptation algorithm named recursive least moduli (RLM) algorithm for use in complex-domain adaptive filters. The algorithm recursively estimates the inverse covariance matrix of the filter reference input, and significantly improves the convergence speed of the least mean modulus (LMM) algorithm. Unlike the RLS algorithm, the RLM algorithm makes the filter highly robust in the presence of impulsive observation noise by introducing "division by error modulus" in the update equations for tap weights and estimate of covariance matrix.

Detailed analysis of the RLM algorithm has been developed to derive a set of difference equations to calculate transient and steady-state convergence behavior in terms of excess mean square error (EMSE). Through experiment, we have demonstrated the effectiveness of the RLM algorithm in realizing fast convergent and robust adaptive filters. Good agreement between simulations and theory has proven the validity of the analysis.

7. REFERENCES

[1] B. Widrow and S. D. Stearns, *Adaptive Signal Processing*. New Jersey: Prentice-Hall, 1984.

[2] S. Haykin, *Adaptive Filter Theory*. New Jersey: Prentice-Hall, 1996.

[3] D. Middleton, "Man-made noise in urban environments and transportation systems: models and measurements," *IEEE Trans. Commun.*, vol. 21, no. 11, pp. 1232-1241, Nov. 1973.

[4] S. Koike, "Adaptive threshold variable step-size least mean modulus algorithm for robust adaptive filtering in impulse noise environments," in *Proc. IEEE ISPACS 03*, Awaji Island, Japan, 7-10 Dec. 2003, C5-1, pp. 586-591.

[5] S. C. Bang and S. Ann, "A robust adaptive algorithm and its performance analysis with contaminated-Gaussian noise," in *Proc. IEEE ISPACS 94*, Seoul, Korea, 5-7 Oct. 1994, pp. 295-300.

[6] V. J. Mathews and S. H. Cho, "Improved convergence analysis of stochastic gradient adaptive filters using the sign algorithm," *IEEE Trans. Acoust., Speech and Signal Processing*, vol. 35, no. 4, pp. 450-454, Apr. 1987.

[7] N. J. Bershad, "Analysis of the normalized LMS algorithm with Gaussian inputs," *IEEE Trans. Acoust., Speech & Signal Processing*, vol. 34, no. 8, pp. 793-806, Aug. 1986.

[8] J. F. Weng and S. H. Leung, "Adaptive nonlinear RLS algorithm for robust filtering in impulse noise," in *Proc. IEEE ISCAS 97*, vol. 4, June 1997, pp. 2337-2340.

[9] M. Abramowitz and I. A. Stegun, *Handbook of Mathematical Functions*. New York: Dover, 1964.