JOURNAL OF ELECTRICAL AND COMPUTER ENGINEERING RESEARCH
VOL. 4, NO. 1, 2024

Performance Comparison of LMS and RLS Algorithms for Ambient Noise Attenuation

Amira Chiheb[1,2,*], Hassina Khelladi[1,2]
[1] Faculty of Electrical Engineering, University of Sciences and Technology Houari Boumediene, Algeria
[2] Materials Physics Laboratory, Faculty of Physics, University of Sciences and Technology Houari Boumediene, Algeria
Emails: achiheb@usthb.dz, hkhelladi@usthb.dz
** Corresponding author*

Abstract: **The aim of this study is to implement two different types of adaptive algorithms for the noise cancellation. The study explores the well-known least mean squares (LMS) adaptive algorithm, which is based on stochastic gradient descent approach, and its performances in terms of noise attenuation level and swiftness in active noise control (ANC). Another algorithm is considered in this investigation based upon the use of the least squares estimation (LSE), commonly named, the recursive least squares algorithm (RLS), and will be compared to the LMS. In order to evaluate the potential of each one, a few simulations are achieved. The numerical experiments are performed by using several real recordings of different environment noises tested on the two proposed adaptive algorithms. A comparison is emphasized regarding noise suppression ability and convergence speed, by implementing both adaptive algorithms on the same noise sources. From this numerical study, the RLS algorithm reveals a faster convergence speed and better control performances than the LMS algorithm.**

Keywords: **Active noise control, FIR filter, LMS algorithm, RLS algorithm.**

I. INTRODUCTION

Noise can be referred as an unwanted signal that can occurs in variety of environments under different forms. One of those types is the acoustic noise that can be emanated from vibrations, colliding sources and moving vehicles [1]. Usually, these noises are absorbed by using passive materials. However, this method is useless and ineffective against low frequencies because the thickness of an absorber must be equivalent or larger than the wavelength of the acoustic noise; this means that an acoustic noise of frequency $100\,Hz$ requires an absorber of width $3.4\ m$, which is impossible to achieve [2].

In order to surpass this problem, another method was initiated named active noise control (ANC). This method is built on cancelling an acoustic noise signal (acoustic pressure) by generating another opposite phase wave, usually named the ″anti-noise″. The sum of the two waves gives the desired silence [2].

The noise is a random process which varies with each instant of time; therefore it requires adaptive filters to generate the ″anti-noise″ wave. Adaptive filters are dynamical systems with variable parameters. They have the ability to change their parameters, using a specific adaptation algorithm, while processing the input signal to maintain an optimal filtering [1].

II. ADAPTIVE FILTERS

An adaptive filter is a filter that can tune its coefficients, using a specific adaptation algorithm, each instance a new sample of the signal is available [3]. The general setup of an adaptive filter is illustrated in Fig. 1, where n is the iteration number. Basically, it receives an input signal $x(n)$ and a desired signal $d(n)$. The filter output $y(n)$ is used to calculate the error signal $e(n)$ as follows [2, 4]:

$$e(n) = d(n) - y(n) \tag{1}$$

The error signal $e(n)$ is then used by the adaptation algorithm to define the most appropriate way to update the filter coefficients [5 – 6].

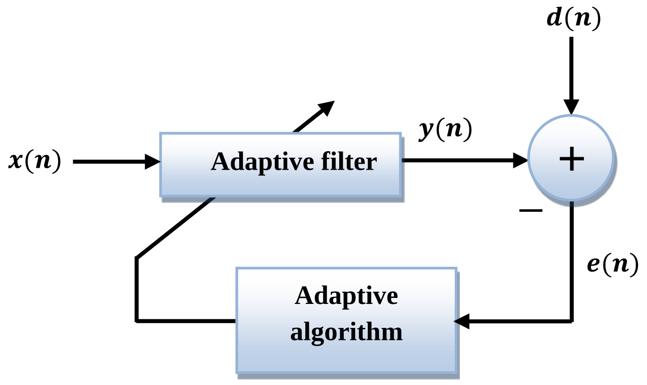


Fig. 1. General adaptive filter system

Adaptive filters can be used in a wide range of configurations. The choice of the structure can influence the computation complexity. Generally, there are two classes of adaptive digital filters, which can be distinguished by their impulse response, commonly named finite impulse response (FIR) filters and infinite impulse response (IIR) filters. In this study, the finite impulse response (FIR) filter is considered [4, 5].

III. FIR FILTERS

Also referred as transversal filter or tapped delay line, is the most used architecture in adaptive digital filtering. It is the most practical choice because of its stability and simple structure that involves a forward paths only (non-recursive structure), as depicted in Fig. 2 [5, 7].

The FIR filter consists of three fundamental features: unit delay element, a multiplier and an adder. The number of delay elements, shown as N in the Fig. 2, is the filter order (number of stages). When z^{-1} operates on $x(n)$, the resulting output is $x(n-1)$. The role of the multipliers is to multiply each tap input $x(n)$ by a filter coefficient w_k (referred as a tap weight).

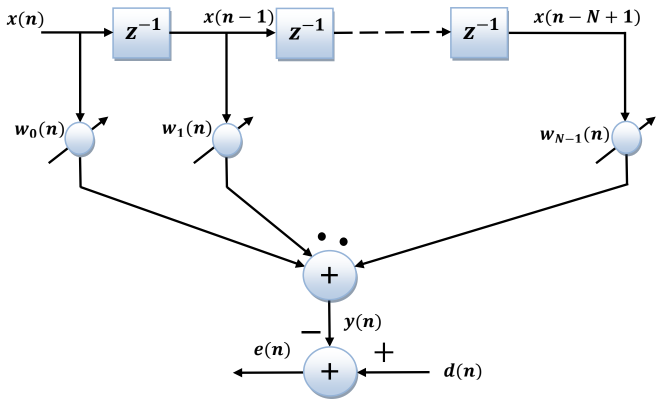


Fig. 2. Adaptive FIR filter structure

The role of the adders is to sum the individual outputs of the multiplier to produce the overall filter response given by [2, 7]:

$$y(n) = \sum_{k=0}^{N-1} w_k(n)x(n-k) = w^T(n)x(n) = x^T(n)w(n) \tag{2}$$

Where: w is an $(N \times 1)$ vector of filter weight coefficients (tap weights) given by :

$$w(n) = [w_0(n) \;\; w_1(n) \ldots\ldots w_{N-1}(n)\,]^T \tag{3}$$

And x an $(N \times 1)$ vector of the input samples (tap input) in the delay chain of the filter, given by :

$$x(n) = [x(n) \;\; x(n-1) \ldots\ldots x(n-(N-1))\,]^T \tag{4}$$

T denotes the transpose of the vectors.

IV. ADAPTIVE ALGORITHMS

The adaptation algorithm allows statistical adjustment of filter weights with the aim of minimizing the cost function (the error signal) [8].

The stochastic gradient and least squares approaches are the most widely used in the development of adaptive algorithms, which led to the development of the most common adaptation algorithms: the LMS and the RLS algorithm [2, 9].

A. Least mean squares algorithm (LMS)

First introduced by Widrow and Hoff in 1960, the LMS algorithm is based on stochastic gradient descent method, since it exploits the gradient vector of the filter tap weight to converge on the optimal solution [1, 6]. It uses the error signal $e(n)$ from previous iterations to update filter weights $w(n)$, then produces an output $y(n)$ that minimizes the mean square error (MSE) [10].

The adaptive filter tap weights are updated using the following formula [2]:

$$w(n+1) = w(n) + 2\mu e(n)x(n) \tag{5}$$

where: n is the number of iterations, $x(n)$ is the input signal, $w(n)$ is the FIR filter's tap weights vector and μ is known as the step size. The selection of an appropriate μ is very important on the performance of the algorithm [1, 9].

The broad range of applications of the LMS algorithm is credited to its low computational complexity, stable behavior with finite precision arithmetic and robustness against different signal conditions [2 - 5].

B. Recursive least squares algorithm (RLS)

This algorithm is based on another approach, called the least squares estimation (LSE). This approach selects the filter coefficients to reduce to the minimum the cost function defined by the sum of weighted error squares for a given data [2, 5, 11]. In recursive implementation of the least squares method, the computation is started with required initial conditions and uses the information contained in new data to update the old estimates. The cost function to be minimized is given by [6, 9]:

$$\xi(n) = \sum_{i=0}^{n} \lambda^{n-i}|e(i)|^2 \tag{6}$$

Where $0 < \lambda < 1$ is the forgetting factor (or weighting factor).

The minimization of this function leads to the formula to update the weight vector, given by the following expression [2, 5, 9, 12]:

$$w(n) = w(n-1) + e(n)K(n) \tag{7}$$

Where: $K(n)$ is the function of forgetting factors [1], also called the gain vector [8 – 9], expressed as [2, 9, 12]:

$$K(n) = \frac{\lambda^{-1}P(n-1)x(n)}{1 + \lambda^{-1}x^T(n)P(n-1)x(n)} \tag{8}$$

$P(n)$ represents the inverse correlation matrix stated by the following formula [2, 9]:

$$P(n) = \lambda^{-1}[P(n-1) - K(n)x^T(n)P(n-1)] \tag{9}$$

Where $x(n)$ is the tap input vector, $w(n)$ is the tap weight vector and n is the number of iterations [9].

This adaptive algorithm presents sophisticated calculations but suffers from a high computational complexity [1, 3, 11].

V. SIMULATION AND PERFORMANCE ANALYSIS

To observe the effectiveness of the two above mentioned algorithms in noise cancellation, a number of numerical experiments are achieved using several recordings of different types of noise. The results are discussed below.

A. Implementation of the LMS algorithm

TABLE I SUMMARY OF THE LMS ALGORITHM

Inputs	Tap weight vector, $\boldsymbol{w(n)}$ Input vector, $\boldsymbol{x(n)}$ Desired signal, $\boldsymbol{d(n)}$
Outputs	Filter output, $\boldsymbol{y(n)}$ Tap weight vector update $\boldsymbol{w(n+1)}$
1. Filtering : $\boldsymbol{y(n) = x^T(n)w(n) = w^T(n)x(n)}$ 2. Error estimation : $\boldsymbol{e(n) = d(n) - y(n)}$ 3. Tap weight vector adaptation : $\boldsymbol{w(n+1) = w(n) + 2\mu x(n)e(n)}$	

The Table I and Fig. 3 summarize the general steps of LMS algorithm implementation.

The process starts with receiving the input signal samples $x(n)$ on the adaptive filter. This later, generates the output signal $y(n)$ and compare it to the desired response $d(n)$, then an error signal $e(n)$ is computed. The error signal is then used to modify and adjust the filter weights vector $w(n)$ in order to reach the optimum solution that minimizes this error as much as possible.

The numerical experiment includes, not only the algorithm implementation by considering two types of noise (jackhammer and an electric saw), but also begins with a basic implementation by applying a sine wave to the algorithm in order to observe the process explained before. The results are displayed in the figures below.

Fig. 4 depicts the process of LMS algorithm. The sinusoidal signal represents the noise to be attenuated and the filter output represents the anti-noise.

Fig. 4(a) shows how the error signal gradually tends towards 0 simultaneously with the generation of the anti-noise that takes on the appearance of the noise signal, demonstrating the principle of (1) in reaching the silence.

The same results are obtained with more complicated noise sources.

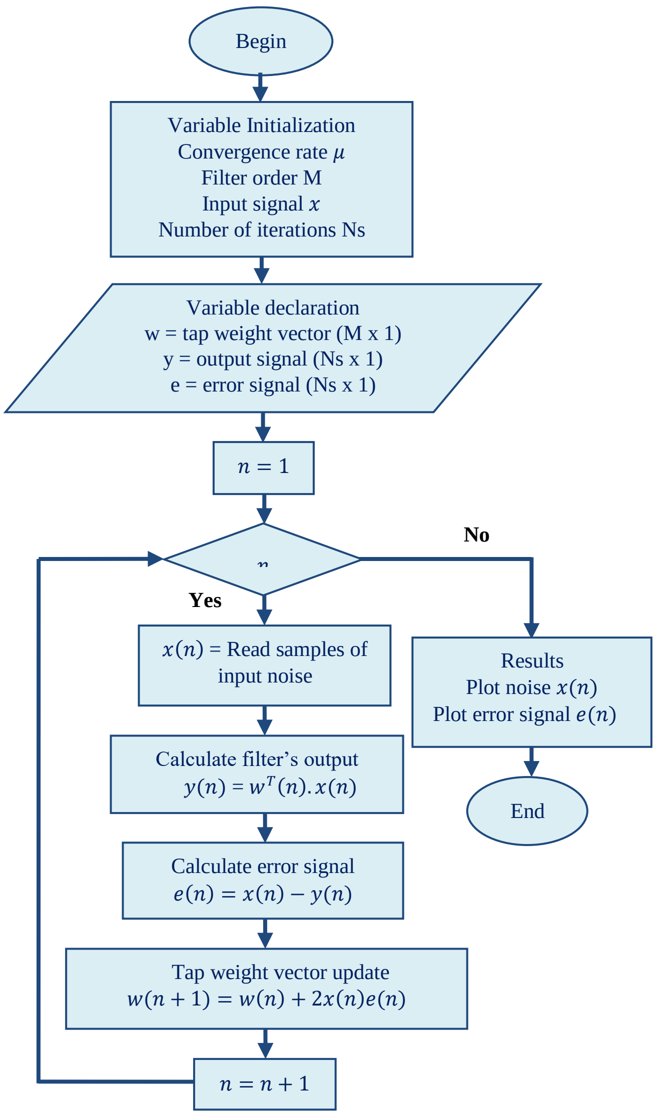


Fig. 3. Flowchart of LMS algorithm

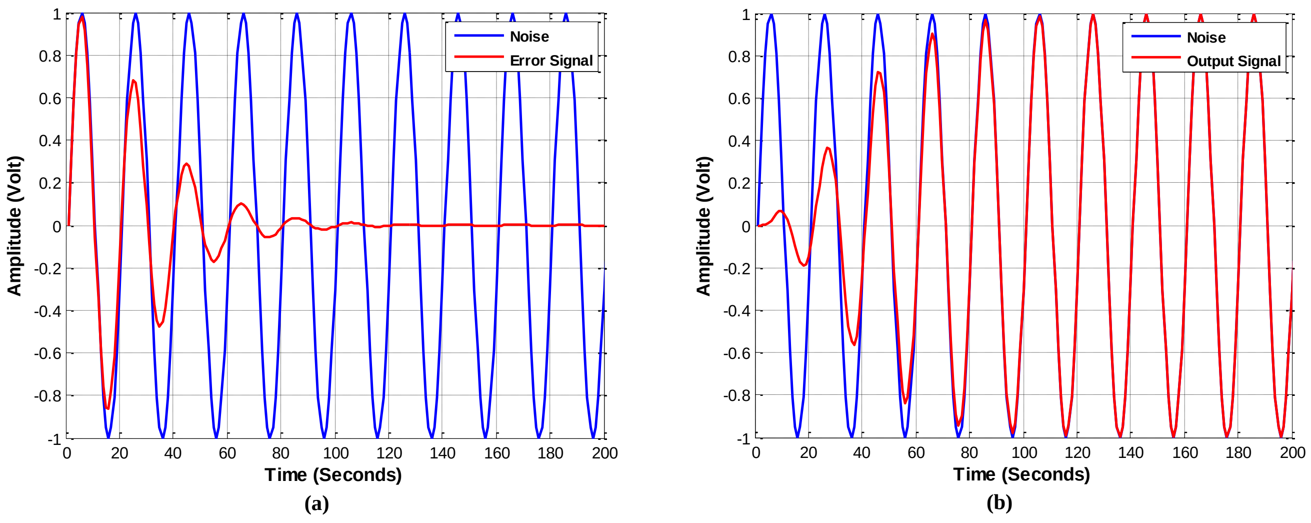


Fig. 4. Implementation of LMS on a sine wave: (a) Error signal, and (b) Output signal.

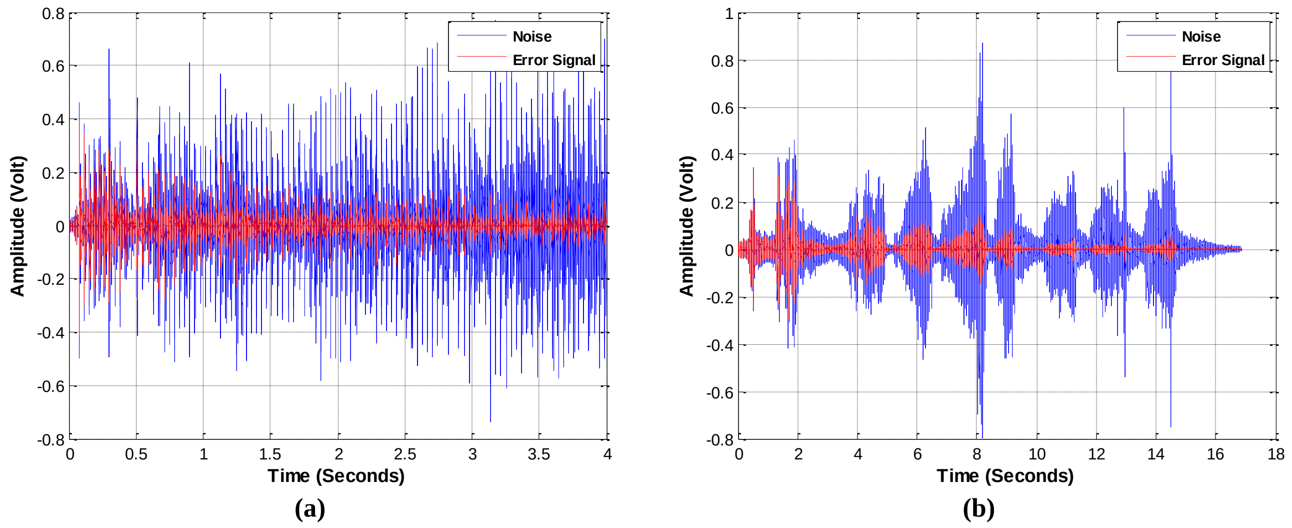


Fig. 5. Implementation of LMS on different noise sources: (a) Jackhammer, and (b) Electric saw.

Referring to Fig. 5, a good attenuation level of the noises is noticed, which means that the LMS algorithm manifests an acceptable performance in noise reduction with the appropriate selection of the convergence rate. In the other hand it reveals a poor convergence speed.

B. Implementation of the RLS algorithm

The same process is repeated with RLS algorithm, where it is implemented to the same noise sources to observe its attenuation capability.

The steps are summarized in Table II, the flowchart of the algorithm in Fig. 6 and the results are displayed in Fig. 7.

TABLE II. SUMMARY OF THE RLS ALGORITHM

Inputs	Tap weight vector estimate $\boldsymbol{w(n-1)}$ Input vector, $\boldsymbol{x(n)}$ Desired signal, $\boldsymbol{d(n)}$ Autocorrelation matrix inverse $\boldsymbol{P(n)}$
Outputs	Filter output, $\boldsymbol{y(n)}$ Tap weight vector update $\boldsymbol{w(n)}$
1. Computation of the gain vector **:** $\boldsymbol{K(n) = \frac{\lambda^{-1}P(n-1)x(n)}{1+\lambda^{-1}x^T(n)P(n-1)x(n)}}$ 2. Filtering : $\boldsymbol{y(n) = x^T(n)w(n-1) = w^T(n-1)x(n)}$ 3. Error estimation : $\boldsymbol{e(n) = d(n) - y(n)}$ 4. Tap weight vector adaptation : $\boldsymbol{w(n) = w(n-1) + K(n)e(n)}$ 5. Update of the autocorrelation matrix inverse : $\boldsymbol{P(n) = \lambda^{-1}[P(n-1) - K(n)x^T(n)P(n-1)]}$	

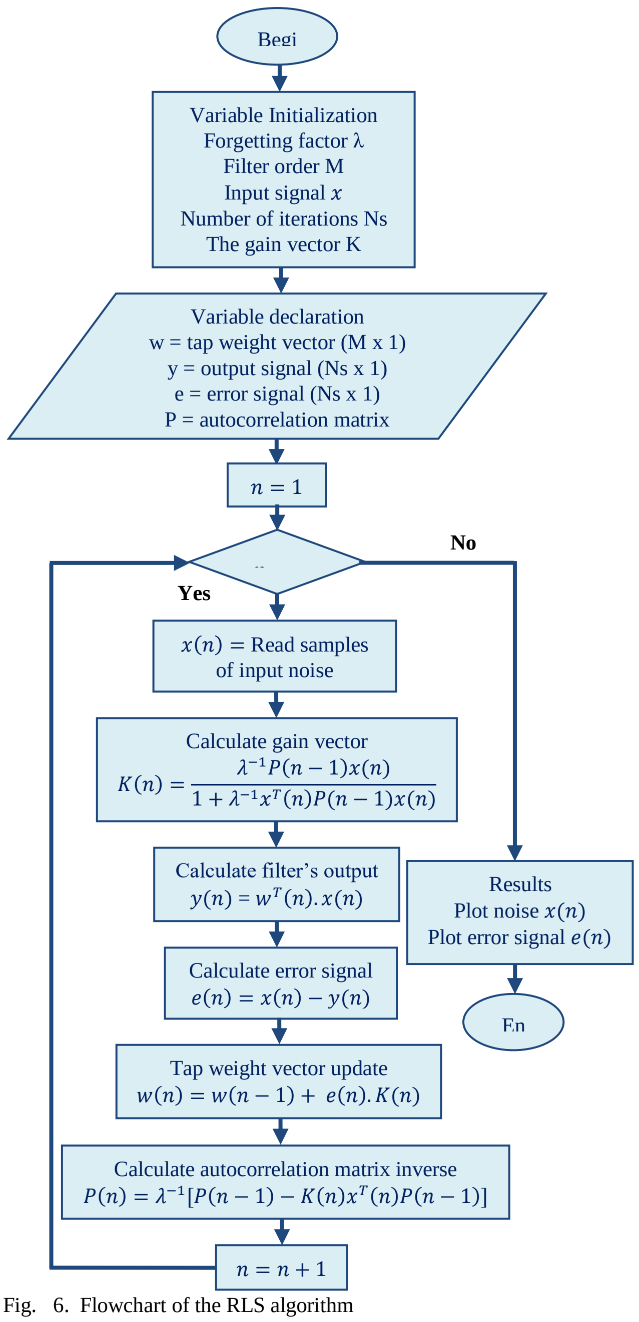
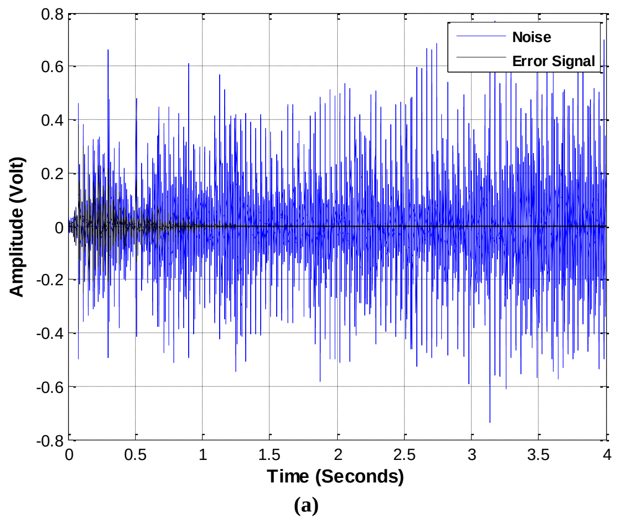

Begi
Variable Initialization
Forgetting factor λ
Filter order M
Input signal x
Number of iterations Ns
The gain vector K
Variable declaration
w = tap weight vector (M x 1)
y = output signal (Ns x 1)
e = error signal (Ns x 1)
P = autocorrelation matrix
$n = 1$
No
Yes
$x(n)$ = Read samples of input noise
Calculate gain vector
$K(n) = \frac{\lambda^{-1}P(n-1)x(n)}{1+\lambda^{-1}x^T(n)P(n-1)x(n)}$
Calculate filter's output
$y(n) = w^T(n).x(n)$
Calculate error signal
$e(n) = x(n) - y(n)$
Tap weight vector update
$w(n) = w(n-1) + e(n).K(n)$
Calculate autocorrelation matrix inverse
$P(n) = \lambda^{-1}[P(n-1) - K(n)x^T(n)P(n-1)]$
$n = n + 1$
Results
Plot noise $x(n)$
Plot error signal $e(n)$
En

Fig. 6. Flowchart of the RLS algorithm



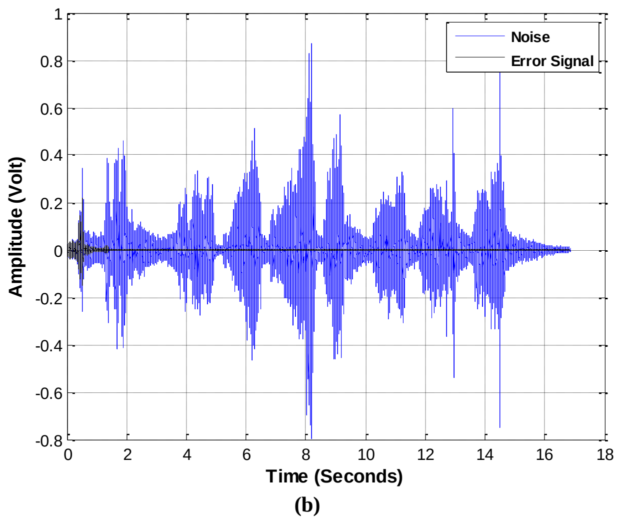


Fig. 7. Implementation of RLS on different noise sources: (a) Jackhammer, and (b) Electric saw.

Referring to Fig. 7, a minimum noise level is reached approximately within 1s, which reveals the fast convergence speed in noise cancellation of the RLS algorithm, and this is due to the appropriate selection of the forgetting factor λ.

C. *Comparison between LMS and RLS algorithms*

In this section, the noise attenuation is displayed for the both algorithms, where the same noise sources are examined. A comparison is established by plotting on the same graph the noise suppression ability of the two proposed techniques.

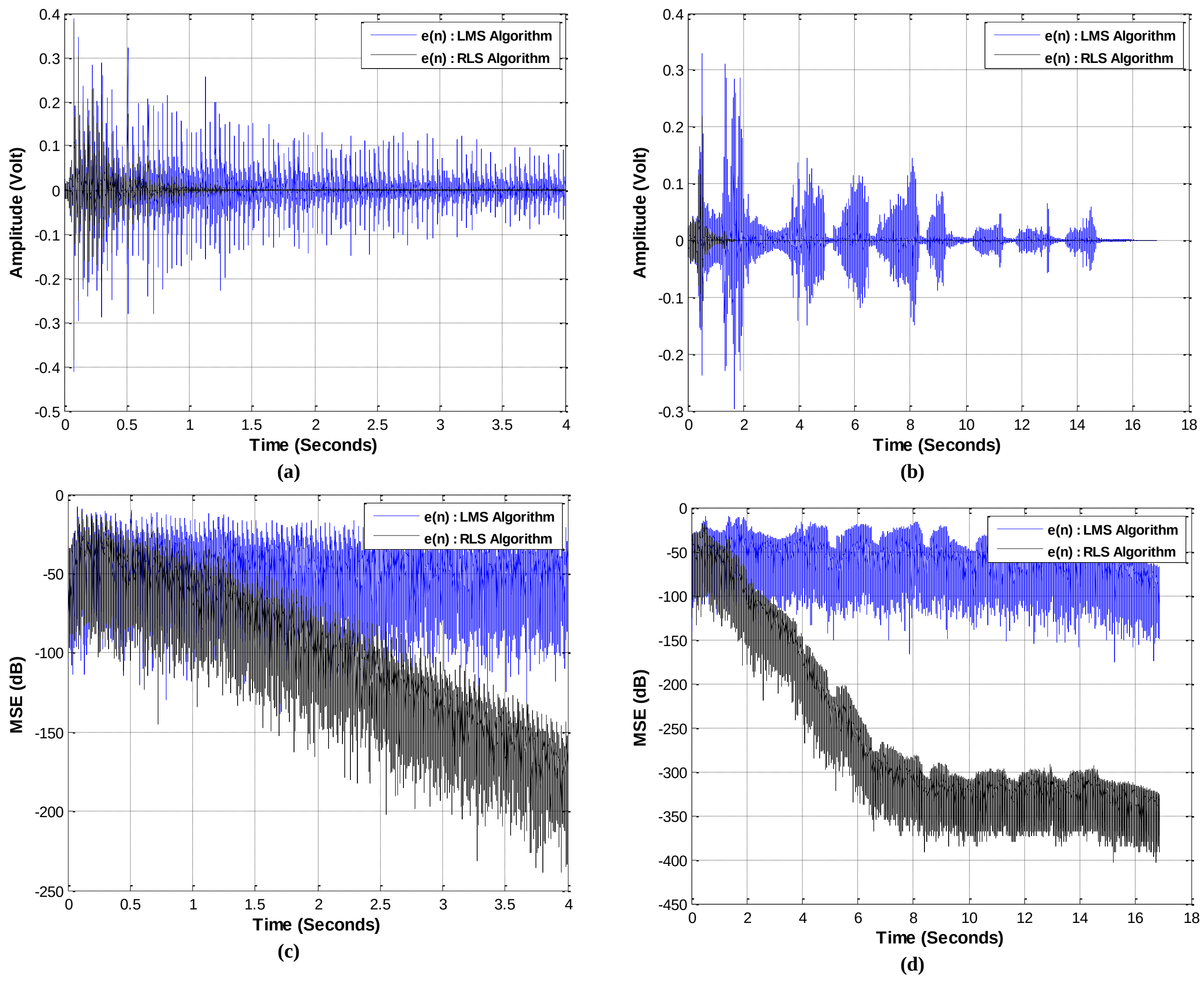


Fig. 8. Comparison of LMS and RLS algorithms implemented on different noises: (a) Jackhammer, (b) Electric saw, (c) MSE in dB for jackhammer, and (d) MSE in dB for electric saw.

An overall observation of Fig. 8(a) and Fig. 8(b) indicates that the RLS algorithm reveals an efficient performance and a faster convergence in attenuating different types of noise, compared to the LMS algorithm. This conclusion is clearly illustrated in Fig. 8(c) and Fig. 8(d), where the mean square error (MSE) is plotted in dB and given by the following equation:

$$MSE(n) = 10\log_{10}(e^2(n)) \tag{10}$$

Referring to Fig. 8(c) and Fig. 8(d) it should be noted that for the first recording, the noise is attenuated by $100\ dB$ for the LMS algorithm while it is about $200\ dB$ for the RLS, and this value is reached after 3 seconds of the first recording. As for the second one, the mean square error is attenuated by $150\ dB$ for the LMS algorithm and by $350\ dB$ for the RLS, and this attenuation is reached after 6 seconds of the second recording, which proves that the RLS surpasses the LMS both in noise attenuation and convergence speed.

VI. CONCLUSION

The attenuation of acoustic noise requires adaptive filters piloted by adaptive algorithms. Those algorithms update the filter tap weights vector to minimize the cost function according to a prescript criterion.

In this work, the noise suppression ability is analyzed for LMS and RLS algorithms with two different types of noise (jackhammer and an electric saw). The results are displayed on different figures to examine the noise attenuation by using each algorithm and a succinct comparison between the two techniques is done.

The numerical experiments reveal good noise attenuation for the two algorithms, while in terms of convergence speed; the RLS algorithm reveals a faster convergence compared to the LMS.

REFERENCES

[1] A. Mon, T.T. Aung and C.H. Lwin, “Active Noise Cancellation in Audio Signal Processing,” *International Research Journal of Engineering and Technology (IRJET)*, vol. 03, no. 11, 2016.

[2] B. Farhang-Boroujeny, *Adaptive Filters Theory and Applications*, Second Edition, John Wiley & Sons, 2013.

[3] L.R. Vega and H. Rey, *A Rapid Introduction to Adaptive Filtering*. Springer Briefs in Electrical and Computer Engineering, 2013.

[4] T. S. Ghmati and A.A.S. Elhoula, “Adaptive Digital FIR Filters: Case Study: Noise Cancellation using LMS Algorithm,” *Albahit Journal of Applied Sciences*, vol. 2, no. 1, 2021.

[5] P.S.R. Diniz, *Adaptive Filtering Algorithms and Practical Implementation*, Fifth Edition, Springer, 2020

[6] T. Lampl, A. Hadi, A. Qadir, J.M.N. Mba Andeme, “Noise cancellation with LMS, NLMS and RLS filtering algorithms to improve the fault detection of an industrial measurement system,” *Research Square*.

[7] C. Hansen, Scott Snyder, Xiaojun Qiu, Laura Brooks, Danielle Moreau, *Active Control of Noise and Vibration*, Second Edition, CRC Press, 2012.

[8] L.V.R. Kumari, A.J. Sabavat and Y.P. Sai, “Performance Evaluation of Adaptive Filtering Algorithms for Denoising the ECG Signal,” *Second International Conference on Electronics and Sustainable Communication Systems (ICESC)*, pp. 1-5, 2021.

[9] S. Haykin, *Adaptive Filter Theory*, Fifth Edition, published by Pearson Education, 2014.

[10] M. Ahmed, A. Farooq, F. Farooq, N. Rashid and A. Zeb, “Power Line Interference Cancellation from EEG Signals Using RLS Algorithm,” *International Conference on Robotics and Automation in Industry (ICRAI)*, pp. 1-5, 2019.

[11] M. Sugadev, M. Kaushik, V. Vijayakumar, I.T. Ilayaraja and K.T. Ilayaraaja, “Performance Analysis of Adaptive Filter Algorithms on Different Noise Sources,” *International Conference on Computer Communication and Informatics (ICCCI)*, pp. 1-5, 2022.

[12] U. Meyer-Baese, *Digital Signal Processing with Field Programmable Gate Array*, Fourth Edition, Springer, 2014.