Jasveen Kaur, Ranjit Kaur / International Journal of Engineering Research and Applications (IJERA) ISSN: 2248-9622 www.ijera.com
Vol. 2, Issue 3, May-Jun 2012, pp.2335-2341

A Comparison Study of Power Spectrum Densities of Various Adaptive Algorithm Using Adaptive Filter

Jasveen Kaur[1] and Ranjit Kaur[2]
[1]Student, Department of Electronics and Communication Engineering
UCoE Patiala-147002, Punjab, India
[2]Associate Professor, Department of Electronics and Communication Engineering
UCoE Patiala-147002, Punjab, India

Abstract – ***A comparison study of different adaptive algorithms(Least Mean Square(LMS), Normalized Least Mean Square(NLMS), Delayed Least Mean Square(DLMS), Recursive Least Square(RLS), QR decomposition based RLS(QRDRLS), Affine Projection(AP)) using Adaptive filter is presented in this paper. The paper focuses on these adaptive algorithms which are used to reduce the noise from the signal which is corrupted by noise. The comparison is done on the basis of power spectrum densities(PSD) after applying these algorithms.***

Keywords: Power Spectrum Density , LMS, NLMS, RLS, QRDRLS, AP

I. INTRODUCTION

Adaptive filter is a self designing filter which means filter performs satisfactorily where complete knowledge about the relevant signal is not known. Adaptive filters therefore have the ability to adjust their own parameters automatically along with their design requires little or no prior knowledge of signal or noise characteristics in order to achieve optimal filtering. Adaptive filter can be divided into linear and nonlinear adaptive filter. Non-linear adaptive has greater signal processing capabilities. Linear adaptive filter is usually used more because non-linear adaptive filter has greater computational complexity[1][2].

As an application noise cancellation can be done using adaptive filter. Noise cancellation can be done by estimating the interference signal and subtracting it from the corrupted signal.

The figure below shows the general adaptive filtering display. The digital filter filters the input signal x(n), to produce the output signal y(n).[3] An appropriate adaptive algorithm is used to adjust the filter coefficient included in the vector w(n), so that the error signal e(n) becomes the minimum. Error is the difference of desired signal d(n) and the filter output y(n). The two noises $N_1(n)$ and $N_2(n)$ are correlated. Therefore adaptive filter is the combination of digital filter and adaptive algorithm.

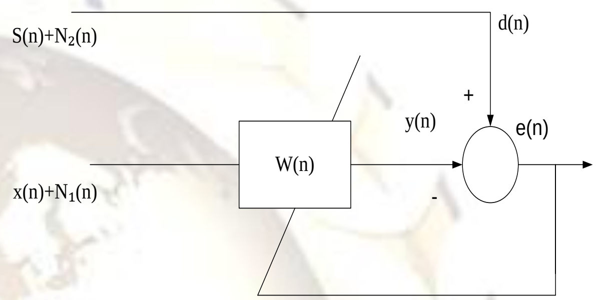

Figure1: Adaptive Noise Cancellation Configuration

II. DESCRIPTION OF ADAPTIVE ALGORITHMS

The adjustment of filter parameters is determined by the adaptive algorithm, its behavior is critical for the filtering performance. So there are different algorithms used to adjust the digital filter coefficients to match the desired response as much as possible.

A. LMS adaptive filter Algorithm:

LMS is an easy algorithm with simple structure, less computation with short filter length. LMS is linear adaptive filtering algorithm[4].

Different implementation steps are given as:

1. Define the desired response and set each coefficient weight to zero.

$$w(n)=0, \quad n=0,1,2,3\ldots\ldots,N \qquad (1)$$

For each sampling instant (n) carry out steps (2) to (4)

2. Move all the samples in the input array one position to the right, now load the current data sample (n) into the first position in the array. Calculate the output of the adaptive filter by multiplying each element in the array of filter coefficients by the corresponding element in the input array then, all the results are summed to give the output corresponding to that data that was earlier loaded into the input array, such that the output y(n) is;

$$y(n) = \sum_{n=0}^{N=1} w(n)x(n) \qquad (2)$$

where x(n) is the input signal and w(n) is the coefficients of the filter

3. Before the filter coefficients can be updated, the error must be calculated, simply find the difference between the desired response, d(n), and the output of the adaptive filter, y(n).

$$e(n) = y(n) - d(n) \qquad (3)$$

4. To update the filter coefficients multiply the error by the convergence step parameter μ, and then multiply the results by the filter input and add this result to the values of the previous filter coefficients.

$$\overline{w}(n+1) = \overline{w}(n) + \mu.e(n).\overline{x}(n) \qquad (4)$$

B. Normalized Least Mean Square(NLMS) Algorithm:

The NLMS is an extension of the standard LMS algorithm and is therefore very similar to it. If we consider the Euclidean norm of the input $\|u(n)\|^2$, the NLMS algorithm with fixed step size ,μ is defined as follows:

$$w(n+1) = w(n) + \mu.e(n).x(n)/ \|u(n)\|^2 \qquad (5)$$

where $w(n+1)$ is the next tap weight value and $w(n)$ is the present tap weight of the adaptive filter.
The step-size parameter μ has to be positive $\mu > O$ and not too large, because otherwise it will cause instability of the LMS algorithm:

$$0 < \mu < 2/\lambda max \qquad (6)$$

where λmax is the largest eigen value of the input signal[6].
NLMS algorithm is the same as the LMS algorithm except the time-varying step size μ(n). This step size generally improves the convergence speed of the filter. The NLMS converges at a faster rate as compared to LMS algorithm.
The drawback of standard LMS algorithm is that it is sensitive to the scaling of its input x(n). This makes it very difficult to choose a step parameter μ that guarantees stability of the algorithm.

C. Delayed Least Mean Square(DLMS) Algorithm:

The filter output y(n), is computed and subtracted from the desired d(n). The error signal e(n) is then used to update the coefficients for the next generation. In some practical applications, the desired signal, is not available until several sampling intervals later. Therefore, there is a delay in the LMS algorithm[5].

The delayed LMS algorithm can be expressed as follows:

$$y(n-\Delta) = w^T(n-\Delta)x(n-\Delta) \qquad (7)$$

$$e(n-\Delta) = d(n-\Delta) - y(n-\Delta) \qquad (8)$$

$$w(n+1) = w(n) + \mu e(n-\Delta)x(n-\Delta) \qquad (9)$$

The delay in the coefficient adaptation has only a slight influence on the steady-state behavior of the LMS algorithm. The delayed LMS algorithm with delay Δ = 1 is widely used in implementing adaptive FIR filtering.

D. Recursive Least Square (RLS) algorithm:

Aiming to minimize the sum of the squares of the difference between the desired signal and the filter output, least square (LS) algorithm could use recursive form to solve least-squares at the moment the latest sampling value is acquired [4]. The filter output and the error function of RLS algorithm is

$$y(n) = w^T(n)x(n) \qquad (10)$$

$$e(n) = d(n) - y(n) \qquad (11)$$

The weighting update equation is

$$w(n) = w(n-1) + \xi(n)[d(n) - w^T(n)x(n)] \qquad (12)$$

Here ξ (n) is the gain coefficient.

According to equation (11) and (12), we can get

$$w(n) = w(n-1) + \xi(n)e(n) \qquad (13)$$

RLS overcomes the disadvantage of LMS algorithm that the convergence speed is slow and it is affected by the input signal. But with high computational complexity, high controller computing and storage performance is needed.

E. QR decomposition based RLS(QRDRLS):

This RLS algorithm recursively updates the estimation for a least squares minimization problem. The computation starts with unknown initial conditions and uses the new data samples to update the old estimation. Rather than dealing with stationary stochastic process, the RLS algorithm will face nonstationary process most of time. Mathematically, RLS algorithm can be described as following a system with a linear combinaer of K-tap[7]. The cost function for this LS minimization problem at time *n* is defined by

$$\varepsilon(n)=\sum_{i=1}^{n}\lambda^{n-i}|e(i)|^2 \tag{14}$$

Where λ is the forgetting factor and has a value of 0< λ<1. *e(i)* is the system error at time instance *i* and is defined by

$$e(i)=d(i)-u^T(i)w(n) \tag{15}$$

d(n) is the desired data at time instance I, w(n) is the tap-weight vector of linear combiner at time instance n and is defined by

$$w(n)=[w_0(n) \quad w_1(n)....w_{k-1}(n)]^T \tag{16}$$

and *u(i)* is the tap-input vector to the linear combiner at time instance *I* and is defined by

$$u(i)=[u_0(i) \quad u_1(i) \quad ...u_{k-1}(i)]^T \tag{17}$$

$$A(n)=[u(1) \quad u(2) \quad u(n)]^T \tag{18}$$

Error is given by:

$$e(n)=d(n)-A(n)w(n) \tag{19}$$

The QR decomposition method has superior numerical stability over the RLS. The QR-decomposition method starts from the data matrix using unitary transformation. For a given unitary matrix **Q(n)**, the cost function can be equivalently expressed as

$$\varepsilon(n)=\left\|Q(n)\wedge^{1/2}(n)e(n)\right\|^2 \tag{20}$$

$$=\left\|Q(n)\wedge^{1/2}(n)d(n)-Q(n)\wedge^{1/2}(n)A(n)w(n)\right\|^2 \tag{21}$$

$$Q(n)\wedge^{1/2}(n)A(n)=\begin{bmatrix}R(n)\\0\end{bmatrix} \tag{22}$$

$$Q(n)\wedge^{1/2}(n)d(n)=\begin{bmatrix}p(n)\\v(n)\end{bmatrix} \tag{23}$$

$$\varepsilon(n)=\left\|\begin{bmatrix}p(n)\\v(n)\end{bmatrix}-\begin{bmatrix}R(n)\\0\end{bmatrix}w(n)\right\|^2 \tag{24}$$

$$\varepsilon(n)=\left\|\begin{bmatrix}p(n)-R(n)w(n)\\v(n)\end{bmatrix}\right\|^2 \tag{25}$$

$$p(n)-R(n)w(n)=0 \tag{26}$$

$$w(n)=R^{-1}(n)p(n) \tag{27}$$

$$Q(n)=Q^{'}(n)\begin{bmatrix}Q(n-1) & 0\\0 & 1\end{bmatrix} \tag{28}$$

Where Q(n) is unitary matrix

$$Q'(n)=G(\theta_{k-1}(n))G(\theta_{k-2}(n))...G(\theta_1(n))G(\theta_0(n)) \tag{29}$$

Where $G(\theta_i(n))$ is the Givens rotation.

Where $\theta_i(n)$ satisfies:

$$\tan(\theta_i(n))=\frac{u_i(n)}{\lambda^{1/2}r_{i,}i(n-1)} \tag{30}$$

F. Affine Projection Algorithm

The affine projection algorithm (APA) [8] is an 'intermediate' algorithm in between the well known NLMS and RLS algorithms, since it has both a performance and a complexity in between those of NLMS and RLS. The APA–recursion for a single desired signal is given as :

$$e(n+1)=d_p(n+1)-X_p^T(n+1)w_{apa}(n) \tag{31}$$

$$g(n+1)=(X_p^T(n+1)X_p(n+1)+\delta I)^{-1}e(n+1) \tag{32}$$

$$w_{apa}(n+1)=w_{apa}(n)+\mu X_p(n+1)g(n+1) \tag{33}$$

$$y(n+1)=x^T(n+1)w_{apa}(n+1) \tag{34}$$

Where p is the number of equations in the system. µ and δ are a step size and a regularization parameter respectively.

III. SIMULATIONS AND RESULTS

On simulation in MATLAB by applying different algorithms we get the following results. Figure 1 to figure 6 shows the sine wave corrupted by noise, the output and the error obtained by applying algorithms. Figure 7 to figure 12 shows the PSD plots before and after filtering.

1)Sine wave corrupted by noise, the Output and the Error plots:

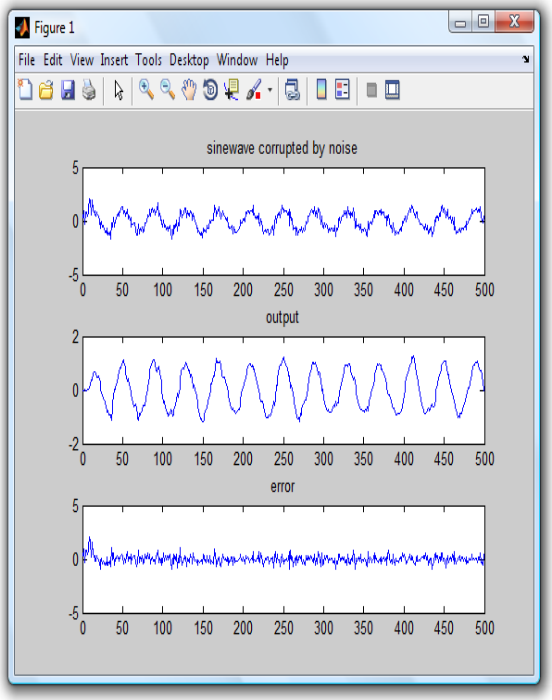

Figure 1. By applying LMS algorithm

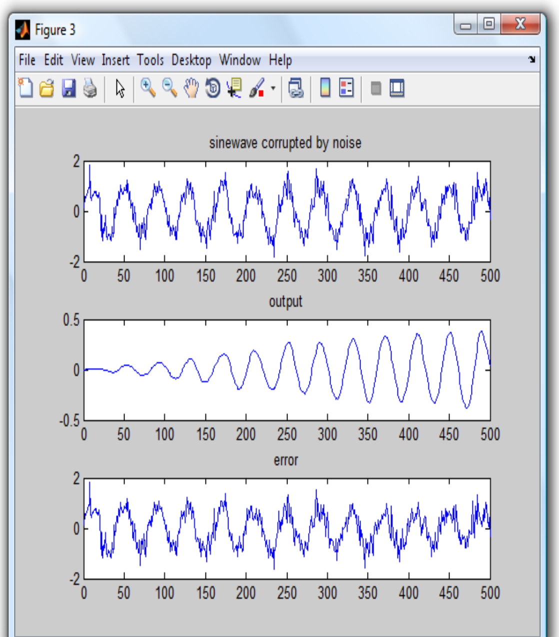

Figure 2. By applying NLMS algorithm

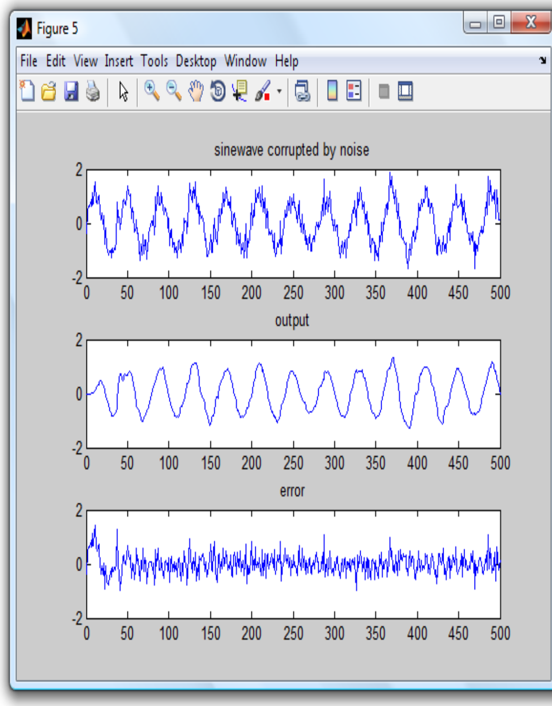

Figure 3. By applying DLMS algorithm

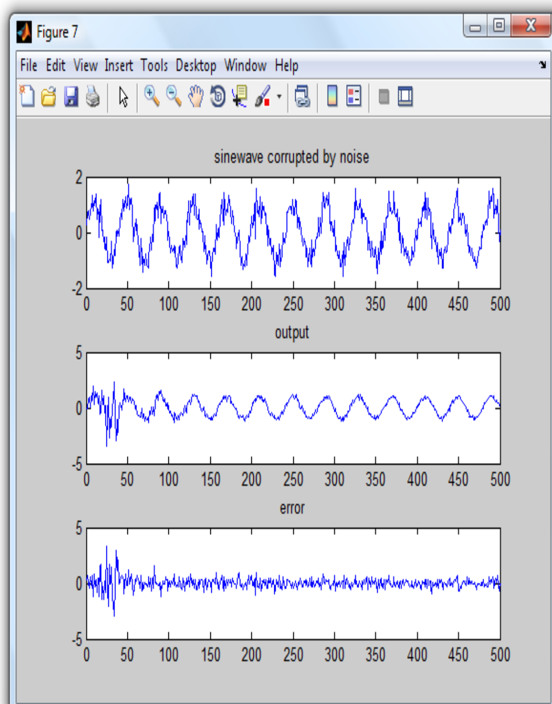

Figure 4. By applying RLS algorthm

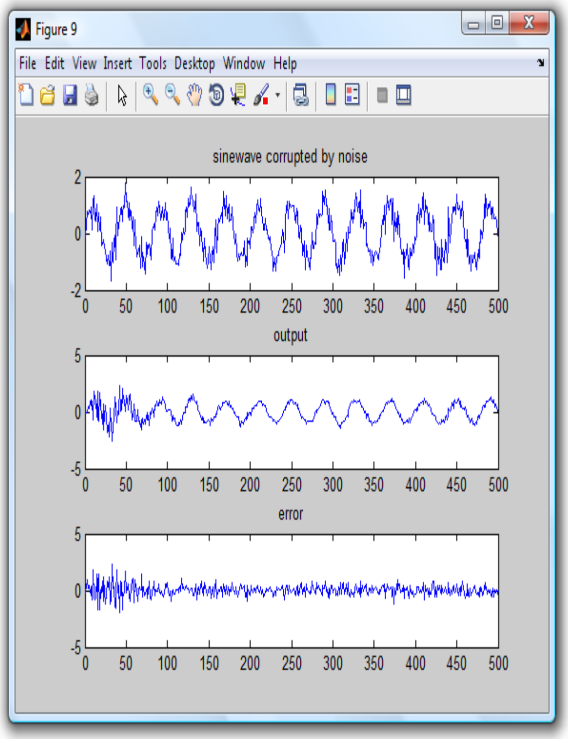

Figure 5. By applying QRDRLS algorithm

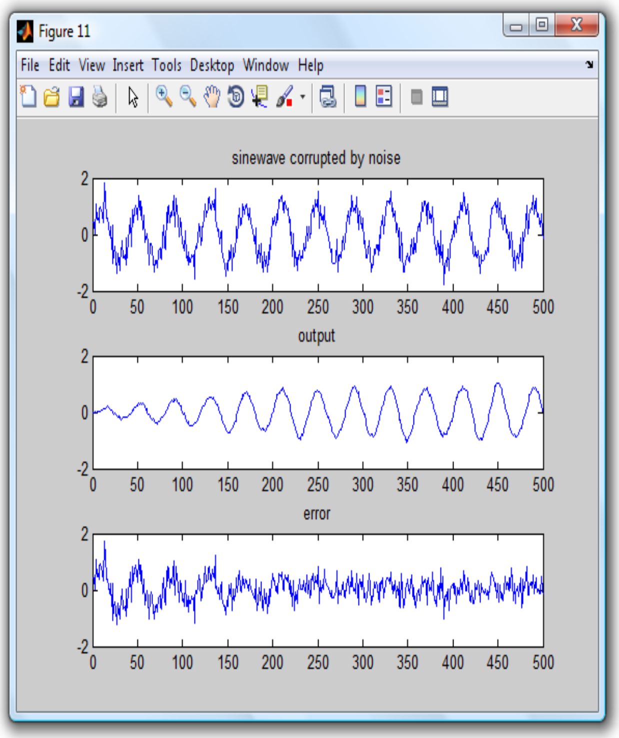

Figure6. By applying AP algorithm

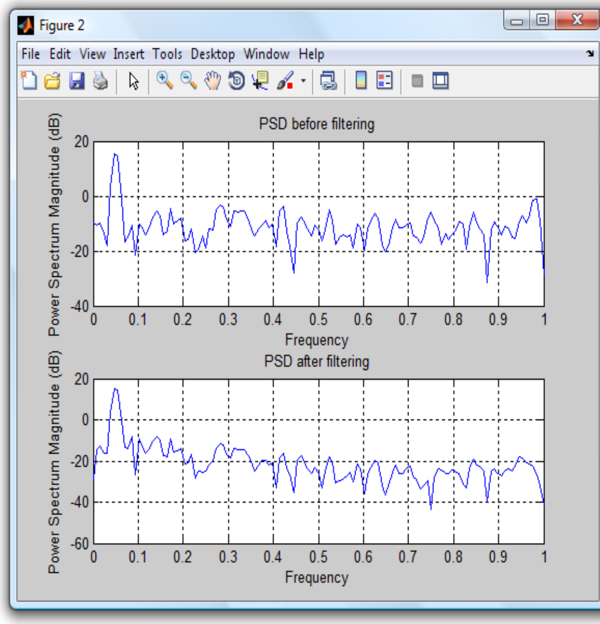

Figure 7. PSD plot before and after applying LMS

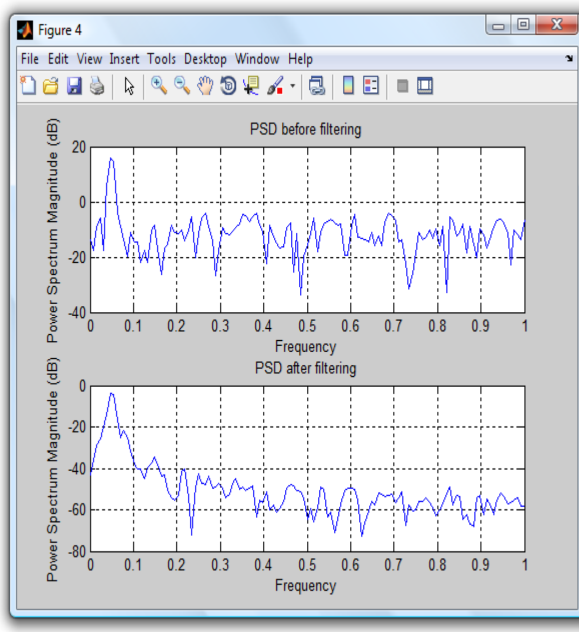

Figure 8. PSD plot before and after applying NLMS

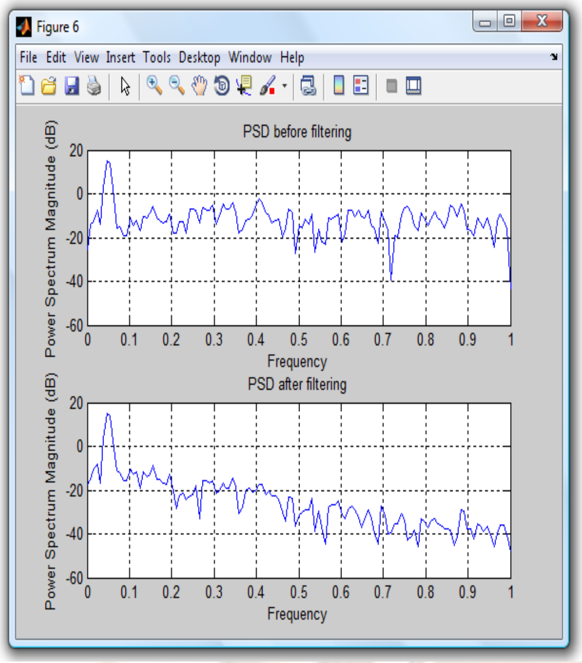

Figure 9. PSD plot before and after applying DLMS

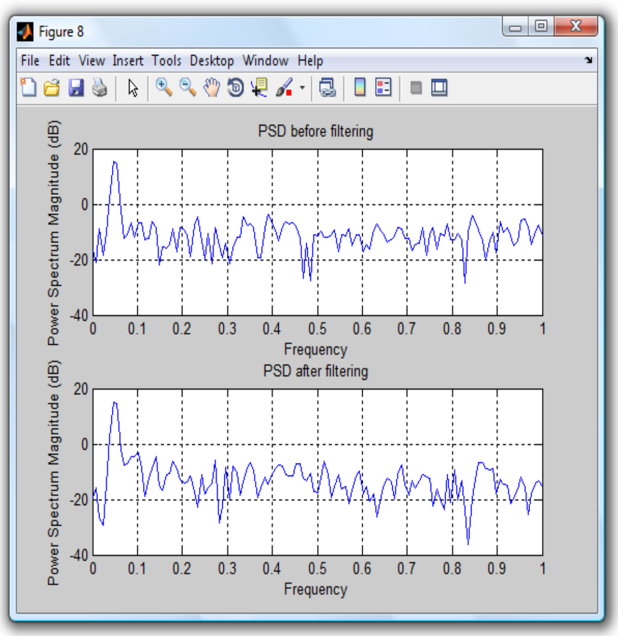

Figure 10. PSD plot before and after applying RLS

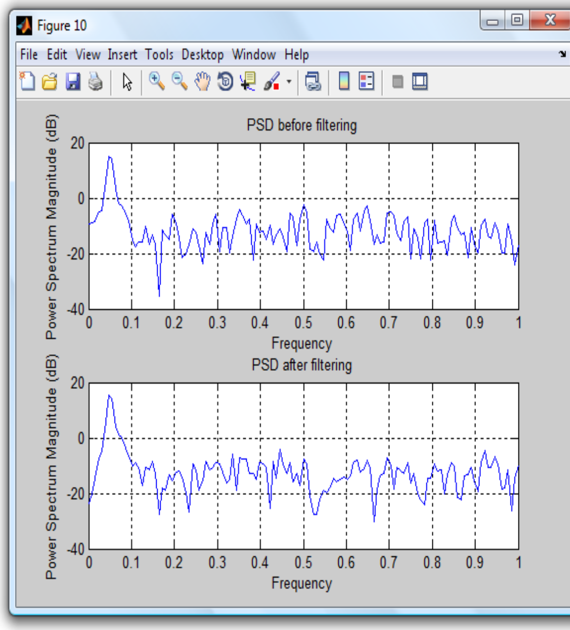

Figure 11. PSD plot before and after applying QRDRLS

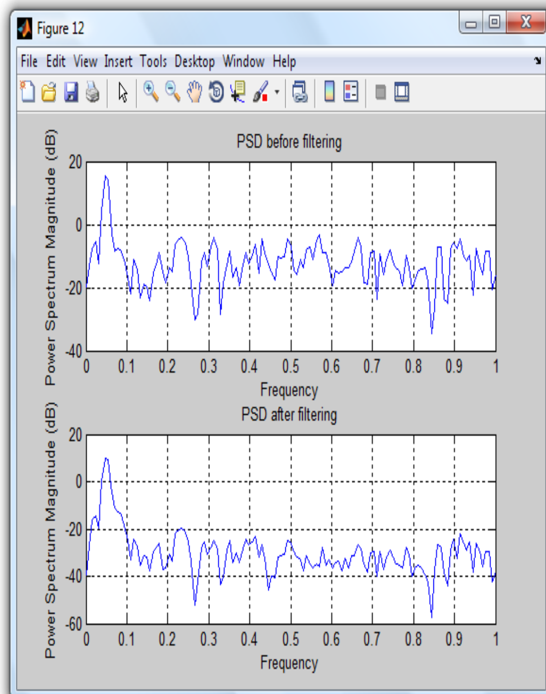

Figure 12. PSD plot before and after applying AP

IV. CONCLUSION

The resulting plots show that the RLS algorithm has a minimum error and NLMS algorithm converges at a faster rate. So AP algorithm is an intermediate between the two algorithms in terms of performance and complexity also.

REFERENCES

[1] Jafar Ramadhan Mohammed, "A New Simple Adaptive Noise Cancellation Scheme Based On ALE and NLMS Filter," University Of Mosul, Electronic Engineering College, Mosul, IRAQ 2007.

[2] Sayed. A. Hadei, "A Family of Adaptive Filter Algorithms in Noise Cancellation for Speech Enhancement," International Journal of Computer and Electrical Engineering, Vol. 2, No. 2, April 2010. 1793-8163.

[3] Ying He, Hong He, Li Li, Yi Wu, "The Applications and Simulation of Adaptive Filter in Noise Cancelling," International Conference on Computer Science and Software Engineering 2008.

[4] Huang Quanzhen, Gao Zhiyuan, Gao Shouwei, "Comparison of LMS and RLS Algorithm for Active Vibration Control of Smart Structures," Third International Conference on Measuring Technology and Mechatronics Automation 2011.

[5] Simon Haykin, "The Principle of Adaptive filter", The electronics industrial publisher, vol. 2, Beijing, 2003, pp.159-398

[6] Ioana Homana, Marina Dana Topa and Botond Sandor kirei, "Echo Cancelling using Adaptive Algorithms," SIITME2009- 15th International Symposium for Design and Technology of Electronics Package.

[7] J. G. McWhirter. Recursive least squares minimisation using a systolic array. In *Proc. SPIE Real Time Signal Processing IV*, volume 431, pages 105–112, 1983.

[8] K. Ozeki and T. Umeda. An adaptive filtering algorithm using an orthogonal projection to an affine subspace and its properties. *Electronics and communications in Japan*, 67-A(5):126 – 132, February 1984.

[9] Georgi Ileiv and Nikola Kasabov, "Adaptive Filtering with Averaging in Noise Cancellation for Voice and Speech Recognition," Deptt. Of Information Science, Univ. of Otago.

[10] Z.D. Yang, Q.T. Huang, J.W. Han, H.R. Li, Adaptive inverse control of random vibration based on the filtered-X LMS algorithm. Earthquake engineering and Engineering Vibration, 2010, 9(1): 141-146.

[11] Colin H Hansen, "Understanding Active Noise Cancellation", Taylor and Francis, 2001