Implementation of LMS FIR based Adaptive Filter for System Identification.

Jatin Yuvraj Pendharkar
Master of Engineering in Information Technology
SRH Hochschule Heidelberg
Heidelberg, Germany
11015240@stud.hochschule-heidelberg.de

***Abstract*—Filters are the most significant instrument in signal processing applications like audio equalizers, picture processing, and so on. As a result, we used an adaptive filter for system identification in our suggested technique. The system must be identified before the filter can be implemented; this is done using Least Men Square (LMS). Because noise is a property that changes over time when a device is operated remotely, we need an adaptation strategy to account for any type of noise that may damage our system and mitigate its impacts in order to provide improved signal quality. As the signal adapts to the original signal, the Mean Squared Error decreases with time. Our proposed algorithm performed according to the desired expectation and can be implemented for System Identification**

Keywords-component; Adaptive Filter, Least Mean Algorithm (LMS), FIR Filter, Learning Curve, Convergence of filter coefficients, System identification.

I. Introduction

Inter symbol interference occurs when a multi-path distortion channel is used in digital communication (ISI). ISI also results in a greater Bit Error Rate (BER). As a result, a filter to compensate for channel distortion must be designed. As a result, adaptive equalizers are increasingly being employed to overcome the channel damage caused by multi-path fading. An adaptive equalization is a linear adaptive filter that is used to represent the channel's inverse transfer function. The algorithm is used to improve the signal to noise ratio of a signal (SNR). Stochastic gradient algorithms and exact least square algorithms are two types of adaptive algorithms. Self-orthogonalizing algorithms are a subtype of stochastic gradient algorithms. Stochastic gradient algorithms include the traditional least mean square (LMS) algorithm (with sample update), normalized LMS algorithm, and block LMS algorithm (with block update). The least squares algorithm is the recursive least squares (RLS) algorithm. The LMS and RLS algorithms are the most often used of these algorithms. The RLS algorithm has a faster convergence rate than the LMS algorithm. However, hardware implementation is quite difficult. Because of its simplicity and resilience, the LMS algorithm is commonly used in hardware implementation. In the sphere of communication, this adaptive equalization technique has been extensively studied and adopted. It's an important element for correcting for a distorted channel. A variable or adjustable system that increases signal processing performance by contacting the outside can be built and is adaptive using this technique. This project employs the use of adaptive filters to reduce noise in digital systems. The behavior of an unknown system was estimated to be a transversal filter in this paper. The extra noise was subsequently removed using adaptive filters and the Least Mean Square technique (LMS)

II. Generalized Block Diagram

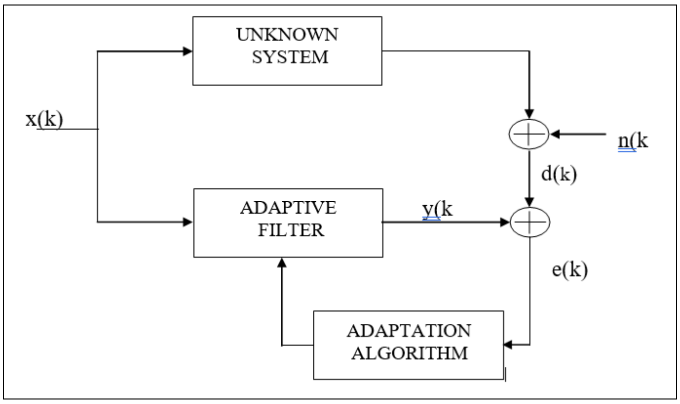

Figure.1. Block Diagram for System Identification.

Signals of interest are frequently contaminated by noise or other signals in the same frequency band in practice. Conventional linear filters can recover the intended signal when the signal of interest and the noise are in separate frequency ranges. Fixed coefficient filters are appropriate when there is spectral overlap between the signal and noise, or when the signal or interfering signal's statistics change with time.

A. Adaption Algorithm

The weight update of an adaptive filter can be done in a variety of ways. The Wiener filter is the best linear filter in terms of mean squared error, and there are numerous algorithms that try to approximate it, such as the steepest descent approach. The least mean square algorithm, created by Widrow and Hoff for use in artificial neural networks, is also available. Other techniques, such as the recursive least squares method and the Kalman filter, are also available. The algorithm chosen is mostly determined by the signals of interest and the operating environment, as well as the required convergence time and computer power.

The least-mean-square (LMS) algorithm modifies the weights by iteratively approaching the MSE minimum, comparable to the Steepest-descent method. Windrow and Hoff invented the LMS algorithm in 1959. To estimate a time-varying signal, the approach employs gradient descent. If a minimum exists, the gradient descent method discovers it by taking steps in the opposite direction as the gradient. It accomplishes this by modifying the filter coefficients to reduce the error. The gradient is the partial derivative of the del operator, which is used to find the divergence of a function, which in this case is the error with regard to the nth coefficient. By taking the negative gradient of a function, the LMS method approaches the function's minimum to minimize error.

B. Transversal FIR Filter..

The FIR filter is implemented serially with a multipliers and adders with feedback as shown in the Figure.2. To reduce saturation, the FIR result is normalized.

The coefficient is updated iteratively by the LMS algorithm and fed to the FIR filter. The FIR filter then generates the output y (n) using this coefficient w_o and the input reference signal x(n). The error generated by subtracting the output y(n) from the desired signal d(n) is used by the LMS algorithm to determine the next set of coefficients.
The elements M-by-1 tap input vector for x(n) is formed by the tap input x(n), x(n-1),....x(n-M+1), where M-1 is the number of delay elements. As the number of iterations n approaches infinity, the value computed for the tap-weight vector w(n) using the LMS algorithm provides an estimate whose predicted value approaches the Weiner solution w_o

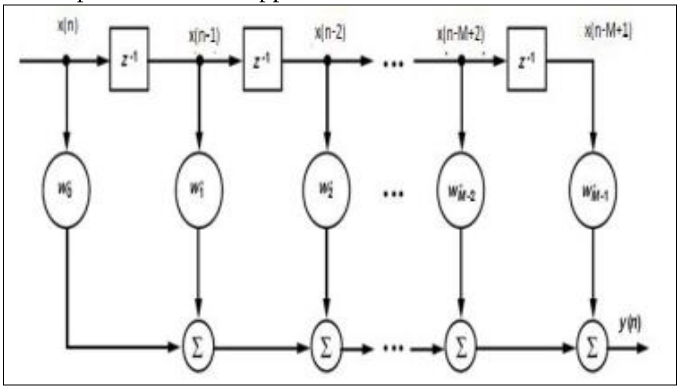

Figure.2. Transversal FIR Filter Implementation

C. System Identification

The adaptive filter's job in this scenario is to discover the inverse model that offers the greatest fit for the unknown plant. The adaptive filter and the unknown plant are linked in a cascade. The adaptive filter transfer function should, in this case, be the reciprocal of the plant's transfer function, so that when both are connected in a cascade, they will be an excellent channel. The input signal drives the plant, while the output of the plant drives the adaptive filter. Equalizers make advantage of this combination of adaptive filters.

System identification, popularly known as mathematical modeling, is a type of adaptive filtering that has a wide range of applications, especially in the field of communication. The adaptive filter is used to generate the linear model that best reflects the unknown plant. The plant and the adaptive filter are connected in parallel and powered by the same input in this design. The plant output is referred to as the system's desired reaction. By subtracting the adaptive filter output from the desired output, the error signal is obtained. The adaptive filter will be time variable or non-stationary if the plant is dynamic.

III. IMPLEMENTATION OF LMS FIR ADAPTIVE FILTER

A. Overview of Operation of LMS Algorithm.

The least mean square algorithm is a linear adaptive filtering technique made up of two steps:

1 A filtering process that entailed computing the output of a transversal filter generated by a collection of tap inputs and generating an estimation error by comparing the output to a desired response.

2. An Adaptive process in which the weight of the filter is automatically adjusted in response to the estimation mistake.

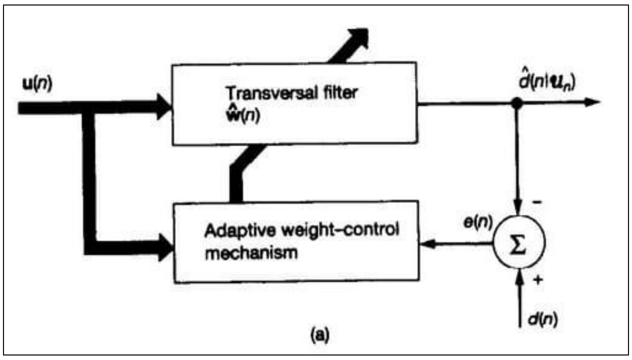

Figure 3. Least Mean Square Algorithm

As a result, the interaction of these two processes creates a feedback loop around the LMS algorithm. First, we have a transversal filter, which is the foundation of the LMS algorithm; this component is responsible for filtering. Second, we have a way for performing adaptive control on the transversal filter's tap weights.

The elements M-by-1 tap input vector for x(n) is formed by the tap input x(n), x(n-1),....x(n-M+1), where M-1 is the number of delay elements. As the number of iterations n approaches infinity, the value computed for the tap-weight vector w(n) using the LMS algorithm provides an estimate whose predicted value approaches the Weiner solution w_o.

The intended response d(n) is submitted for processing during the filtering procedure, along with the input vector x(n). The transversal filter gives an output that is utilized as an approximation of the intended response, d, given this input (n). As a result, the difference between the desired response and the actual filter output can be defined as an estimation error e(n).

Adaptive Weight-control Mechanism. For k = 0,1,2, 3,…M-2,, M-1. It is the scalar of the product of estimation error e(n) and input x(n)The correction made to the tap weight at iteration n+1 is defined by the result so achieved. The scaling factor used in this computation is denoted by μ. It is called the step size parameter

When comparing the LMS algorithm's control mechanism to the technique of steepest descent, we can observe that the LMS employs the product of x(n-k) e*(k) as an estimate of element k in the gradient vector ∇J(n), whereas the method of steepest descent utilizes the product of x(n-k) e*(k). As a result, the gradient noise affects the recursive computation of each tap weight in the LMS algorithm. The steepest descent approach generates a tap weight vector w(n) that goes down the ensemble-averaged error performance surface on a deterministic trajectory, eventually arriving at the wiener solution w0. The existence of gradient noise, on the other hand, causes the LMS algorithm to behave differently. Rather than stopping at the wiener solution, the tap weight vector performs a random motion around the minimum point of the error performance surface.

We previously mentioned that the LMS algorithm uses feedback in its functioning, which posed the issue of stability. In this case, a useful criterion would be to demand that.

$$J(n) \rightarrow J(\infty) \text{ as } n \rightarrow \infty$$

where J(n) is the mean squared error generated by the LMS algorithm at time n, and J (∞) is the final value. In the mean square, an algorithm that meets this criterion is said to be convergent. The step size μ. parameter must satisfy a criterion related to the eigen structure of the correlation matrix of the tap input for the LMS algorithm to satisfy this criterion.

B. LMS Algorithm

The LMS algorithm uses the gradient vector estimates from the available data. LMS includes an iterative process that successively corrects the weighting vector in the direction of the negative of the gradient vector, which ultimately leads to the smallest mean square error. Algorithms The LMS algorithm is relatively simple; it does not require calculation of the correlation function, nor does it require matrix inversions. LMS algorithms have a step size that determines the amount of correction to be applied when the filter adapts from one iteration to the next. Adaptive filter design skill is required to choose the suitable step size.

The process by which the error signal (the difference between the output signal and the target signal) reaches an equilibrium state over time is known as filter convergence.

The LMS algorithm, which was started with an arbitrary weight vector value, converged and remained stable for a long time.

$$0 < \mu < 1/\lambda_{max}$$

Where λ_{max} is the largest eigenvalue of the correlation matrix R. The algorithm's convergence is inversely proportional to the correlation matrix R's eigenvalue spread Convergence may be slower when R's eigenvalues are widely distributed. The correlation matrix's eigenvalue spread is calculated by dividing the biggest eigenvalue by the smallest eigenvalue of the matrix. If μ is set to a small value, the algorithm will converge slowly. A higher value of μ may speed up convergence, but it may be less stable near the minimum value. Based on multiple estimations, one of the literatures also provides an upper bound for μ <= 1/(3trace(R)).

The LMS algorithm can be summarized by following equations

$$y(n) = w^{h*} x(n)$$

y(n) is the output of the LMS Filter. It is a scalar product of the input coefficient and the transpose of the filter weight vector coefficients.

$$e(n) = d^*(n) - y(n)$$

e(n) is the error between the desired output d(n) and the actual output from the filter y(n).

$$w(n+1) = w(n) + \mu\, x(n)e^*(n)$$

Where w(i+1) is the updated weight vector which is the summation of the weight of optimal filter wo and product of the step size or convergence parameter μ with input signal vector and error signal

C. Learning Curve

In the Learning curve, the adaption of the filter coefficient w was examined and the behavior of the error over time. During the transient (learning) phase, the error signal is not stationary. The mean square error depends on K the number of iterations.

Stationary mean Square Error of LMS.

$$J = J_{min} + J_{ex}$$

J ex: is the excess error which is only in LMS is dependent on the step size μ

If the size of μ is not desirable then it causes variation in the mean square error.

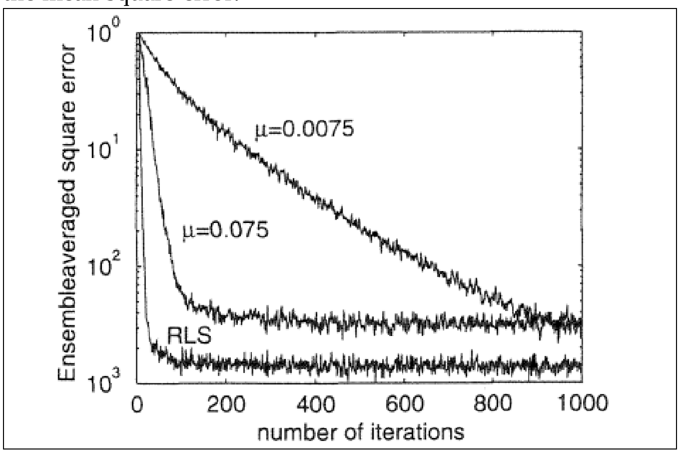

Figure 4. Learning curve

In the figure 4. The μ = 0.075 gives the excess error (J ex) in the LMS and μ = 0.0075 gives the J min minimum square error for number of iterations = 1000.

IV. RESULTS AND GRAPHS

A. Task 1. LMS Algorithm

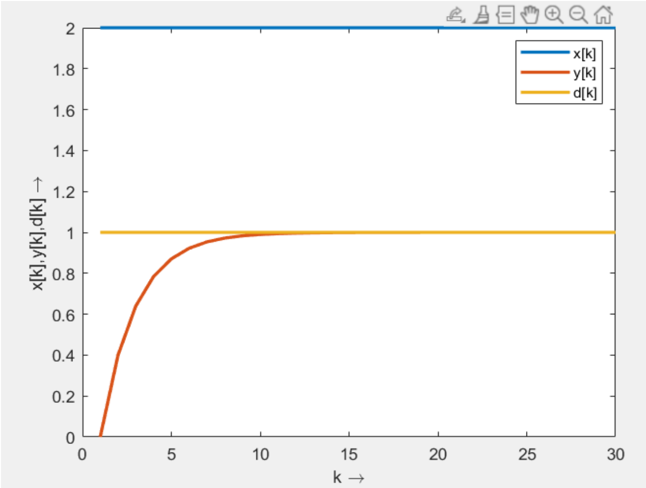

Figure 5. Output of LMS Algorithm

For x = 2,d=1, and k=1,2,3….999 and mu= 0.01
Here, we get output graph as the scalar product of X(n) and Transpose of Filter coefficient Wo.The graph consist of x(k) input signal,Desired signal d(k) and output signal y(k).

B. Task 2. Learning Curve.

- Task 2.1

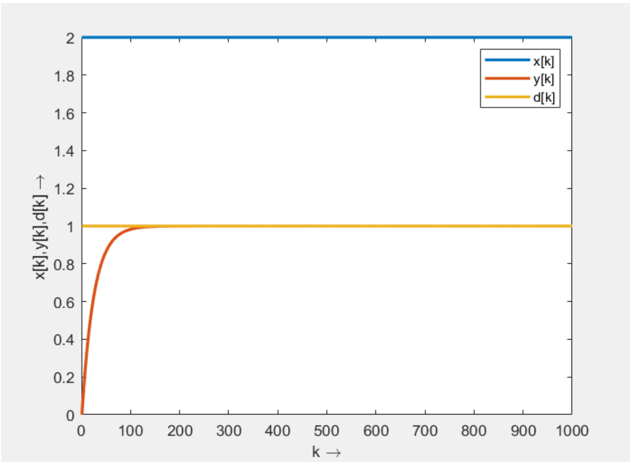

Figure 6. Output for LMS Algorithm for using in Task2.

For x=2,d=1, k= 1,2,3….999 and mu =0.01
Here, we get output graph as the scalar product of X(n) and Transpose of Filter coefficient Wo. The graph consist of x(k) input signal,Desired signal d(k) and output signal y(k).

- Task 2.2

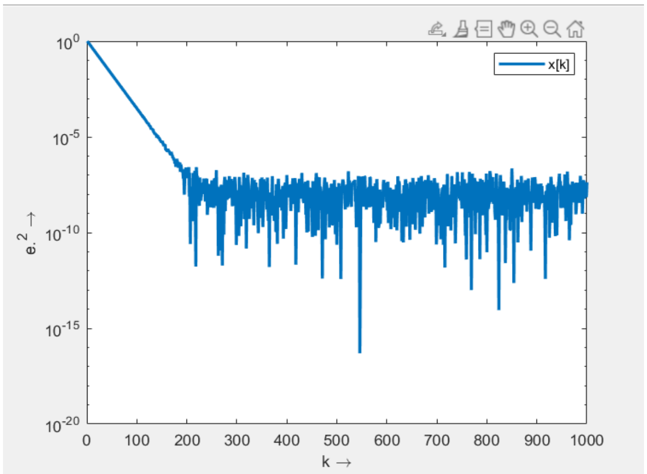

Figure 7.Output for learning curve with noise.

For x = random signal with noise, d = d+ noise ,mu = 0.01 and k=1,2,3….999.
Here, we get output graph as the scalar product of X(n) with random noise added to see how it effects the output and Transpose of Filter coefficient Wo.The graph consist of x(k) input signal,Desired signal d(k) and output signal y(k).We can easily plot from the graph that there is a lot of noise in the output signal which might lead to loss of important signal data.

- Task 2.3

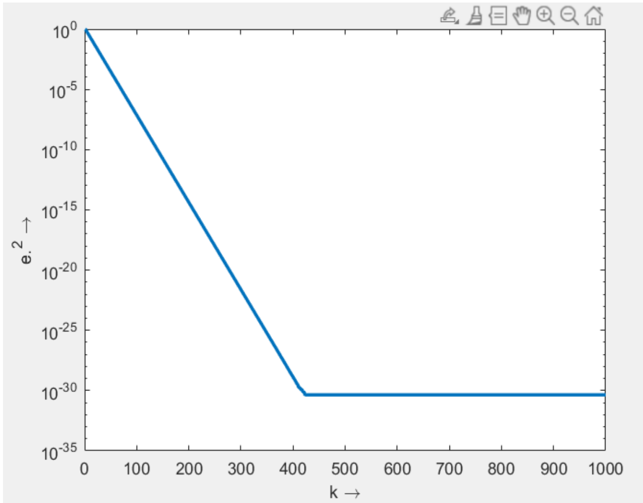

Figure 8. Output for Error e.^2 without Noise

For x= random signal and d = 1, mu= 0.01 and k= 1,2,3,….999
Here, we get output graph as the scalar product of X(n) without random noise and Transpose of Filter coefficient Wo.The graph consist of x(k) input signal,Desired signal d(k) and output signal y(k).We can easily plot from the graph that there is no noise signal in the output signal of the learning curve.

- Task 2.4

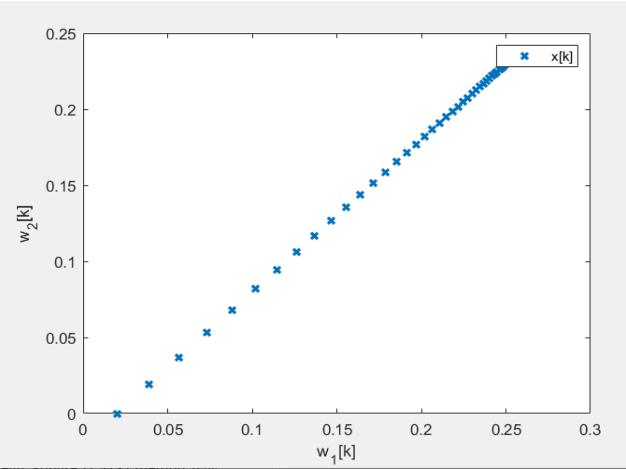

Figure 9. Convergence of filter coefficient.

Here we get output for the convergence by plotting the Graph with reference to the w1[k] and w2[k].Where we can plot that as the iteration increase we can see that we see convergence in the filter coefficient as it gets the new value of w[k] by iterations being carried out. $W(k) = [w1[k],w2[k]]^T$.

C. Task 3. System Identification

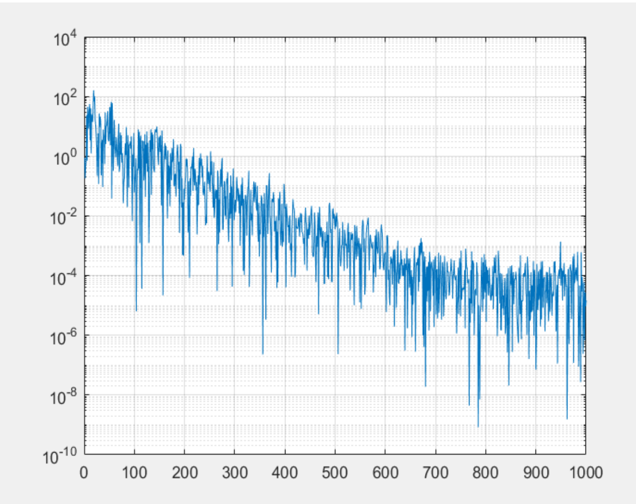

Figure 10. Output for System Identification for N=4.

Here we have taken input by loading the '09_task3_x_d.mat' file which had values of x (input signal) and d(Desired Signal),We chose N=2 and mu = 0.01. For achieving the system Identification, it is necessary to choose the value of the mu and N precisely.

Conclusion

The project results show that the adaptive filter correctly estimates and converges on the unknown system coefficients. The effect of step size and number of iterations on filter performance, such as mean square error and estimate accuracy, has been widely analyzed. The reduction in step size reduces steady state error while also increasing the convergence time. Initially, the prediction process was carried out using the Least Mean Square (LMS) method, which identified the system based on its input and output responses. The prediction differed from the original signal by an acceptable amount. The following is the outcome of the analysis: The prediction procedure was initially carried out using the Least Mean Square (LMS) method, which identified the system based on its input and output responses. The prediction was within a reasonable range of the original signal. The analysis yielded the following results: The Least Mean Square (LMS) method was used to identify the system based on its input and output responses in the beginning of the prediction phase. The prediction matched the original signal within a tolerable range. The step size variation has a significant impact on the performance of the LMS algorithm. The step size determines the amount of correction to be applied to the input signal at each iteration in order to achieve the desired output. When considering a system with a sine wave as the desired signal, it is found that if the step size is small, the algorithm takes more iterations to reach the desired signal. As a result, we can deduce that if the step size is very small, the system convergence will necessitate a greater number of iterations. The system's convergence rate is faster if the step size is kept very large, but the system becomes unstable, which is not expected, thus the step size is chosen so that it is neither too little nor too large.

References

[1] S.D.Shinde, Mrs.A.P.Patil, ,Mrs.P.P.Belagali3 ,"SYSTEM IDENTIFICATION USING ADAPTIVE FILTER ALGORITHMS," IOSR Journal of Electronics and Communication Engineering (IOSR-JECE) ISSN: 2278-2834-, ISBN: 2278-8735, PP: 54-59.

[2] B.Swapna Reddy, V. Rama Krishna on Implementation of Adaptive Filter Based on LMS Algorithm International Journal of Engineering Research & Technology (IJERT) ISSN: 2278-0181Vol. 2 Issue 10, October - 2013I.

[3] Ravi H Bailmare, S.J.Honale, Pravin V Kinge, "Design and Implementation of Adaptive FIR filter using Systolic Architecture" International Journal of Current Engineering and Technology, E-ISSN 2277 – 4106, P-ISSN 2347 - 5161

[4] Ali Akbar Siddiqui, S Hassan Ahmed,Habib Shaukat, M. Shafiq and Dr. M. Tahir Qadri on "Real Time Adaptive Filter Implementation For System Identification And Noise Elimination" Journal of Information & Communication Technology, Vol. 9, No. 2, (Fall 2015) 42-54

[5] Ibraheem Kasim Ibraheem, on "Adaptive System Identification Using LMS Algorithm Integrated with Evolutionary Computation" in Electrical Engineering Department, College of Engineering, Baghdad University

[6] Saurabh R. Prasad, Bhalchandra B. Godbole, on "Optimization of LMS Algorithm for System Identification" , DKTE Society's Textile and Engineering Institute, Ichalkaranji, India

[7] Ms.Kinjal Rasodia, Dr. Kiran Parmar on "Adaptive Filter Analysis for System Identification Using Various Adaptive Algorithms," in International Journal of Engineering Research & Technology(IJERT), ISSN: 2278-0181, Vol. 1. 3rd May 2012.

[8] Todd K. Moon, Wynn C. Stirling, "Mathematical Methods and Algorithms for Signal Processing .

[9] Simon Haykin, "Adaptive Filter Theory,Third Edition" J. Name Stand. Abbrev., in press..