

## Review of Computer Engineering Research

2020 Vol. 7, No. 2, pp. 73-85.

ISSN(e): 2410-9142

ISSN(p): 2412-4281

DOI: 10.18488/journal.76.2020.72.73.85

© 2020 Conscientia Beam. All Rights Reserved.



# LMS ALGORITHM FOR ADAPTIVE TRANSVERSAL EQUALIZATION OF A LINEAR DISPERSIVE COMMUNICATION CHANNEL

 Farrukh Arslan

School of Electrical and Computer Engineering, Purdue University, West Lafayette, USA.

Email: [farslan@purdue.edu](mailto:farslan@purdue.edu) Tel: 001-765-4301-437



## ABSTRACT

### Article History

Received: 21 July 2020

Revised: 24 August 2020

Accepted: 10 September 2020

Published: 1 October 2020

### Keywords

Adaptive signal processing  
Adaptive transversal filter (ATF)  
Adaptive equalization  
Least mean square (LMS)  
algorithm.  
Simulation  
MATLAB.

The presence of any type of distortion in communication system, regardless of the causes, is undesirable and undeniably has a negative impacts on the system in general and therefore it is necessary to eliminate its effects. This study employs one of the well-known algorithms for adaptive equalization of linear dispersive communication channel which is Least Mean Square (LMS) algorithm. The LMS technique is basically utilized to eliminate the noise in communication channel. The novelty of this paper includes the profoundly analyzing of the influence of rate of convergence, miss-adjustment, computational requirement, and sensitivity to Eigen-value spread in sufficient details in a simple and plain way. Moreover, the system performance improvement employing the feedback equalizer technique is intensively presented which shows that our methodology is very effective to eliminate the noise in the system. The simulation work has been performed with MATLAB software.

**Contribution/Originality:** This study uses new estimation methodology which is regarded as profoundly analyzing of the influence of rate of convergence, miss-adjustment, computational requirement, and sensitivity to Eigen-value spread in sufficient details in a simple and plain way.

## 1. INTRODUCTION

Adaptive filters are used extensively in statistical signal processing and offer a great improvement in performance compared with the conventional fixed filters [1-7]. The subject of adaptive filters in general and linear adaptive filters in particular has drawn the attention of many researchers and therefore a various methodologies have been developed and implemented to solve any given problem in the area of statistical signal processing. The linear adaptive filter includes a filter whose function is to produce a desired output, and an adaptive algorithm to set the filter parameters. The selected algorithm is significantly affected by the filter structure which is mainly classified into finite impulse filter (FIR) [8-14] and infinite impulse response (IIR) [15-19].

Generally, adaptive algorithm attempts to minimize the error function in the input, reference, and output signals to near zero value. The most commonly minimization methods used for adaptive filters are Quasi-Newton techniques and the steepest-descent gradient technique [20, 21]. The latter is easy to perform but the quasi-Newton strategy basically has better convergence rate. Therefore the best choice is the Quasi-Newton techniques which have better computational performance and good convergence. But disadvantage of this method is very sensitive to the instability matters. In all these strategies, it is necessary to select the convergence factor carefully based on the specific adaptation issue. The error signal normally is created in different ways but the most popular

techniques are Mean Square Error (MSE) methodology, and Least Squares (LS) technique. MSE is requiring an infinite amount of data. The LS technique is consistent with the fixed data. Proper selection of the error signal basically impacts the selected algorithm complexity, and convergence rate.

Any adaptive application has to be carefully studied prior to selection of the adequate algorithm. The selection of algorithm must consider the computational cost, performance, and robustness. In this applied study we select the LMS algorithm [22-25] rather than the other two well-known algorithms namely, Recursive Least Squares RLS and Recursive Least Squares Lattice RLSL algorithms that could be employed to solve problems related to the field of equalization. The main objective of using this strategy is to eliminate noise from the corrupted input signal and adapt the ATF weights in the way that the mean square of the estimated error is to be minimized. Upon applying the algorithm to the linear equalization we can study the various aspects, behaviours, advantages, and drawbacks of the technique. Moreover this technique, which is essentially employed in the field of adaptive filters, has many different applications in the fields of communications, computers and adaptive signal processing in general [26-30] due to its computation simplicity [31].

Figure 1 represents the block diagram of the channel equalization. A binary data  $a(n)$  is transmitted with random values of (+1), and (-1) having variance  $\delta = 1$  and zero mean. This type of data is produced by random noise generator (1) and transmits through the channel, which has the following impulse response:

$$b(n) = \begin{cases} \frac{1}{2} \left[ 1 + \cos\left(\frac{2\pi}{\omega}(n-2)\right) \right] & \text{for } n = 1, 2, 3 \\ 0 & \text{otherwise} \end{cases}$$

Both distortion and eigenvalue spread are controlled by tap weight vector ( $w$ ). Random generator [2] generates white noise  $v(n)$  which has zero mean and  $\delta^2 = 0.0001$  and by this we get SNR of 40 dB at the input of filter. It is worthy to know that both noise generators are independent of each other.

The best tap weights of the filter are symmetric around mid-point. If the filter taps are selected to be 11, so the best tap-weights will be 6 and since the  $b(n)$  is symmetric around time  $n=2$ , and the data transferring starts at 1, then the input  $\{a_n\}$  is delayed by  $1+5 = 6$  samples to deliver the best response. The convolution sum of  $a(n)$  and  $v(n)$  yields the equalizer input  $u(n)$ :

$$u(n) = b^T a(n) + v(n); n=1, 2, \dots, N$$

$$u(n) = b_1 a_{n-1} + b_2 a_{n-2} + b_3 a_{n-3} + v(n)$$

$$\text{Where } a_{n-i} = 0; n-i < 0$$

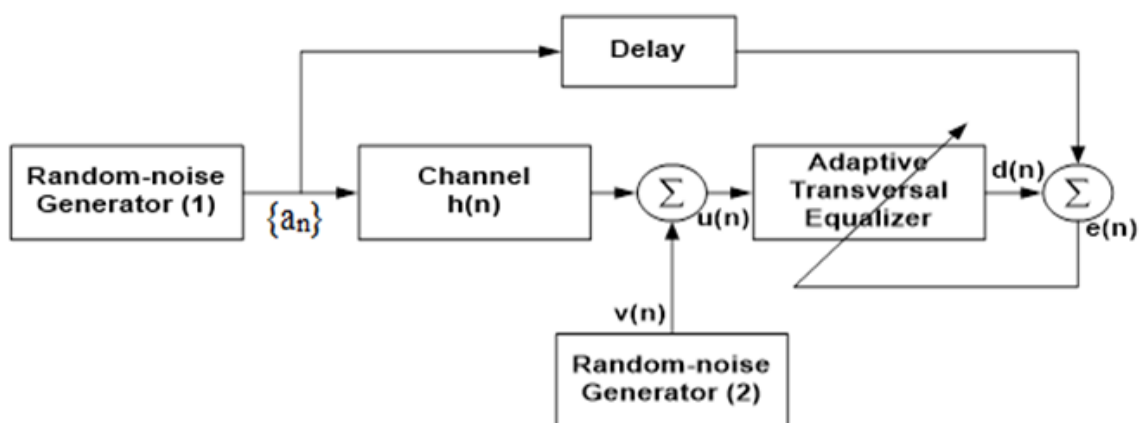


Figure-1. Block diagram of the typical channel equalizer.

## 2. PROPOSED LMS ALGORITHM

### 2.1. Problem Statement

It is necessary to eliminate the effects of distortion produced in the transmitting communication channel so as to produce the desired signal  $d(n)$  throughout the updating of the ATF weight. In this algorithm, which is the simplest one, we calculate the estimated error  $\{e(n)=y(n) - d(n)\}$ . This error is employed to update tap weight vector "w" values as follows:

Step 1: Select an initial weight vector, for example,  $w(0) = 0$

Step 2: For each sample of the input sequence  $\{u(n)\}$ ,  $n = 1, 2, \dots, N$ , form the tap-input vector  $u(n)$ , and compute the adaptive transversal ATF output  $y(n) = w^T(n-1) u(n)$ .

Step 3: Calculate the error  $e(n) = y(n) - d(n)$

Step 4: Update  $w(n) = w(n-1) + \mu u(n) e(n)$

Step 5: Go to Step 2 until  $n = N$ .

Figure 2 depicts the flow chart for explaining the LMS algorithm.

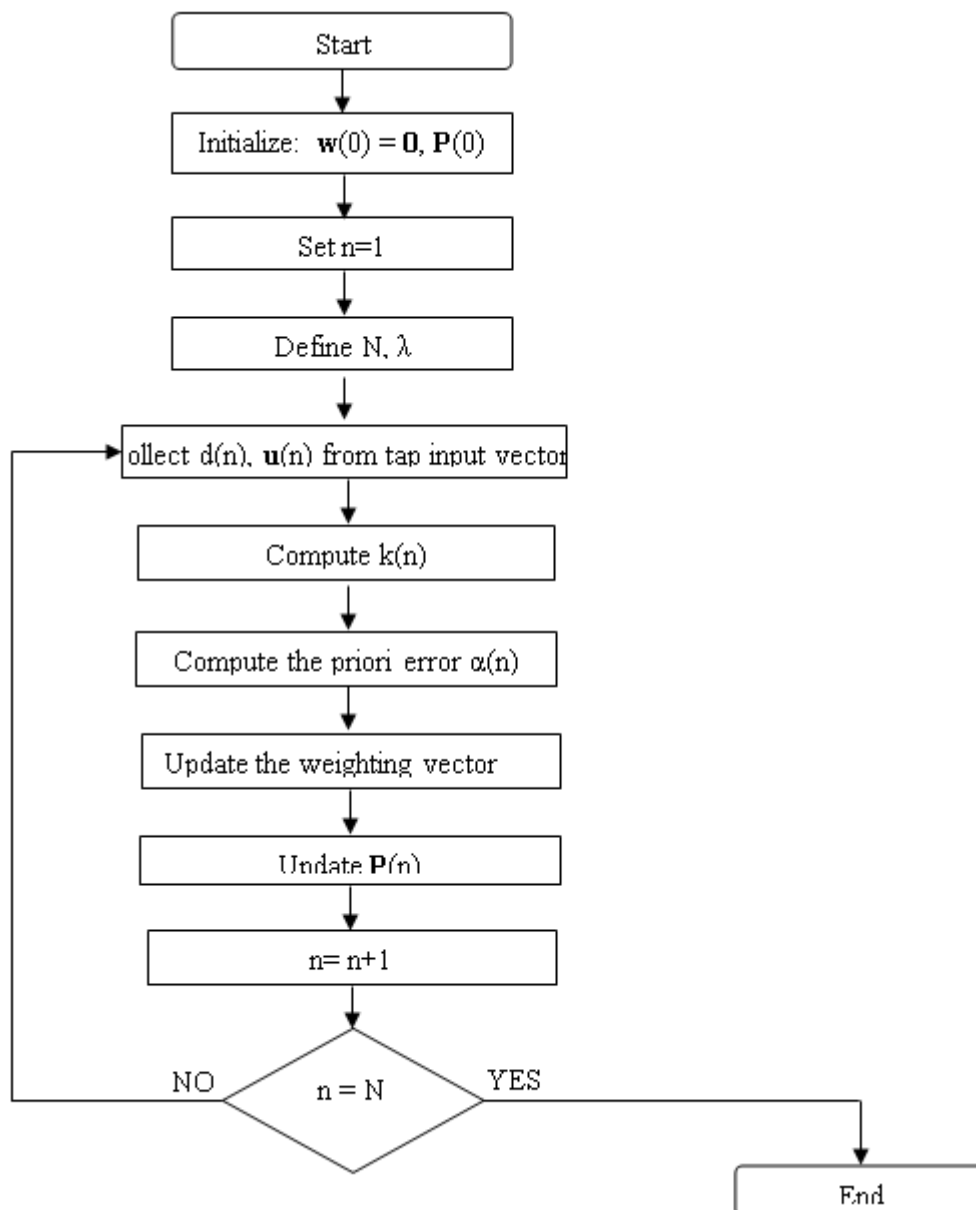


Figure-2. Flow chart for LMS algorithm.

The mean square error is then computed to study the characteristics of this simple technique. So, by applying this algorithm, we investigate the effect of ATF design, eigenvalue spreads, step-size parameter ( $\mu$ ), and finally the effect of decision feedback equalizer.

### 3. RESULTS AND DISCUSSION

#### 3.1. Effect of Adaptive Transversal Filter Order

Table 1 shows the step-size parameter and eigenvalue spread used for both filters.

Table-1. Parameters used to select filter size.

w	$\lambda$	SNR	$\mu$
3.3	21.7132	40 dB	0.07

Figure 3 demonstrate the learning curves of two ATF sizes  $M=11$ , and  $M=21$  for a channel with  $w=3.3$  (corresponds to eigenvalue spread of 21.7132),  $SNR=40$ dB, and  $\mu=0.07$ . According to the second order analysis, the step-size parameter has to be less than  $(2/Mr(0))$ . The value of  $r(0)$  corresponds to  $w=3.3$  is 1.2265 therefore  $\mu$  shall be less than 0.148. We conclude that the value of  $\mu=0.07$  is appropriate to the ATF size  $M=11$ , and the averaged square error decreases with the increasing number of iterations and reaches steady state after iteration 300, but the case is different with adaptive transversal ATF order  $M=21$ , because the step size is so high so that the averaged square error is in ascending order. If we select proper step size  $\mu$ , such as 0.035 and apply this for both ATF orders as shown in Figure 4, we come to know that the difference between both curves is insignificant. Therefore, based on this result we select the ATF order  $M=11$ . This selection is consistent with the fact that design of any system shall be cost effective, so it is not reasonable to select higher ATF order.

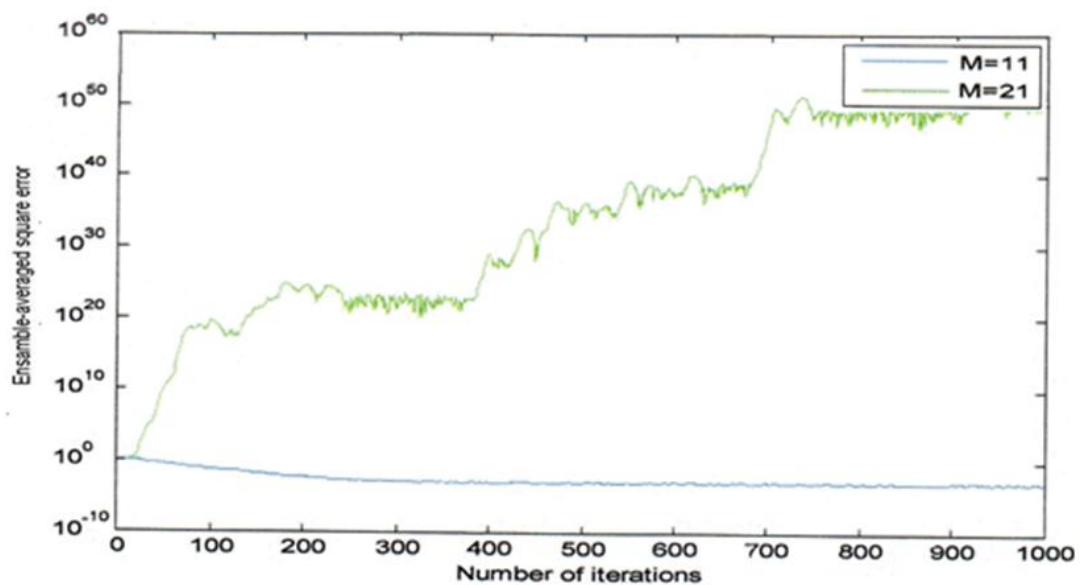


Figure-3. Curves of LMS algorithm for filter of taps  $M=11$ , and  $M=21$ , with  $\mu=0.07$ ,  $w=3.3$ ,  $SNR=40$  dB.

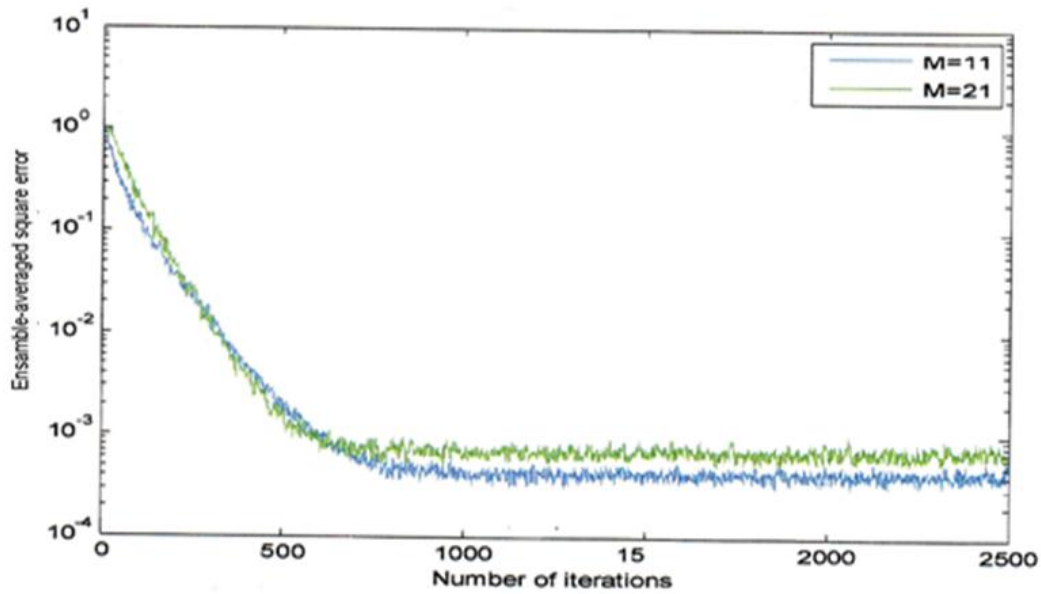


Figure-4. Curves of LMS algorithm for filter of taps M=11, and M= 21, with  $\mu = 0.035$ ,  $w = 3.3$ , SNR=40 dB.

### 3.2. Effect of Eigenvalue Spread

In this study, the step-size parameter is kept fixed at  $\mu = 0.07$ .

Where;  $\mu < \frac{2}{\text{tr}[\mathbf{R}]} = \frac{2}{\text{tr}[\mathbf{R}]}$  (for second order analysis).

For selected M=11, the autocorrelation (R) will be a symmetric matrix of size 11\*11. It is given that b(n) has nonzero values only for n=1,2,3, so the only nonzero elements in the matrix are r(0), r(1), r(2).

$$R = \begin{bmatrix} r(0) & r(1) & r(2) & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ r(1) & r(0) & r(1) & r(2) & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ r(2) & r(1) & r(0) & r(1) & r(2) & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & r(2) & r(1) & r(0) & r(1) & r(2) & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & r(2) & r(1) & r(0) & r(1) & r(2) & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & r(2) & r(1) & r(0) & r(1) & r(2) & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & r(2) & r(1) & r(0) & r(1) & r(2) & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & r(2) & r(1) & r(0) & r(1) & r(2) & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & r(2) & r(1) & r(0) & r(1) & r(2) \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & r(2) & r(1) & r(0) & r(1) \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & r(2) & r(1) & r(0) \end{bmatrix}$$

Now, the procedure to calculate r(0), r(1), r(2), for each value of w is as follows:

We know that  $r(k) = E[u(n)u(n-k)]$ , where  $k = 0, 1, 2, \dots, M-1$ .

Hence  $r(0) = E[u(n)u(n)]$ ;  $r(1) = E[u(n)u(n-1)]$ ;  $r(2) = E[u(n)u(n-2)]$

Substituting the value of u(n) in the equation:

$$u(n) = b_1 a_{n-1} + b_2 a_{n-2} + b_3 a_{n-3} + v(n)$$

$$\text{So, } r(0) = E[b_1 a_{n-1} + b_2 a_{n-2} + b_3 a_{n-3} + v(n)]^2$$

But it is evident that;

$$E[a_{n-1} a_{n-k}] = \begin{cases} 1, & i=k \\ 0, & i \neq k \end{cases}$$

$$E[v_{n-1} v_{n-k}] = \begin{cases} 1, & i=k \\ 0, & i \neq k \end{cases}$$

Hence

$$r(0) = b_1^2 + b_2^2 + b_3^2 + \delta_v^2$$

Similarly:

$$r(1) = b_1 b_2 + b_2 b_3, \text{ and } r(2) = b_1 b_3$$

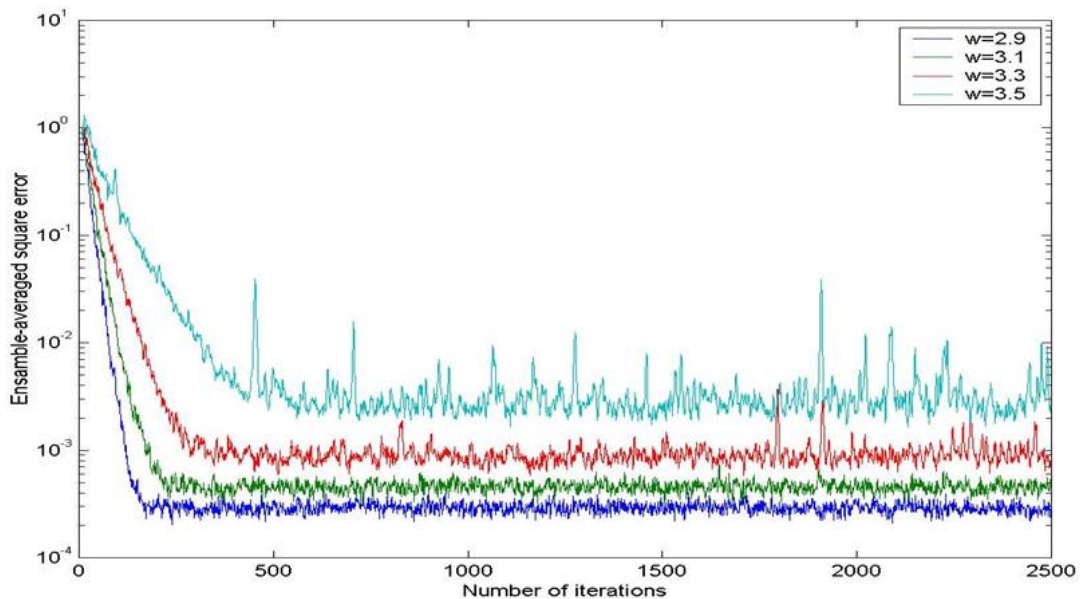
The following table shows the effect of change of the distortion parameter ( $w$ ) on the elements of autocorrelation matrix ( $R$ ) and eigenvalue spread.

**Table-2.** The auto-correlation matrix  $R$ , eigenvalue spread  $\chi(R)$  for varying channels.

	$w=2.9$	$w=3.1$	$w=3.3$	$w=3.5$
$r(0)$	1.0964	1.1567	1.2265	1.3023
$r(1)$	0.4388	0.5596	0.6729	0.7775
$r(2)$	0.0481	0.0783	0.1132	0.1511
$\chi(R)$	6.07.82	11.1238	21.7132	46.8216
$2/ Mr(0)$	0.1658	0.1571	0.1482	0.1396

The appropriate value of  $\mu$  must be selected from this equation  $\mu < \frac{2}{Mr(0)}$ , and must guarantee the convergence for all channels mentioned in the above table.

The value of  $2/Mr(0)$  has been calculated for each channel as in Table 2 so, value of ( $\mu$ ) is to be less than the worst case ( $w=3.5$ ) which is 0.1396. The step size (0.070) is appropriate for all channels. Figure 5 depicts the learning curves for various channels with fixed ( $\mu$ ) =0.07.



**Figure-5.** Curves of LMS algorithm for various channels of  $\mu=0.07$ , SNR=40 dB.

It is clear from the above Figure that as the eigenvalue spread ( $\chi$ ) increases, the convergence speed of adaptation process will decrease associated with the increasing of the ensemble-averaged square error. This indicates that the LMS strategy is very sensitive to  $\chi$ . It is important that the eigenvalue spread ( $\chi$ ) is related directly to ( $w$ ).

Figure 6 below depicts the optimum tap-weights values obtained after iteration (2500) for each of the four eigenvalues spread. It can be seen that the tap weights of the equalizer for all four eigenvalues are symmetric around 6 since the ATF order is 11. The value of center-tap increases with increasing of the eigenvalue spread and this leads to the conclusion that the change in eigenvalue will affect the impulse response of the ATF.

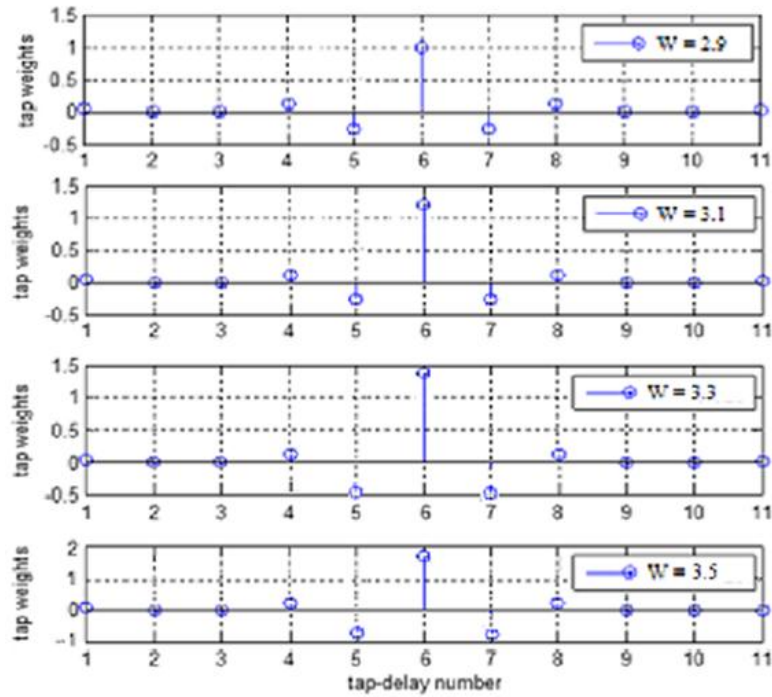


Figure-6. The equalizer impulse response for different channels using LMS algorithm.

### 3.3. Effect of Step-Size Parameter

Figure 7 depicts the curves of the LMS algorithm for a fixed eigenvalue spread and varying  $\mu$ . The value of ( $w$ ) is kept constant at 2.9 while ( $\mu$ ) takes the values (0.07, 0.025, and 0.0075]. It is crystal clear that the rate of convergence is dependent on the ( $\mu$ ) and the smaller ( $\mu$ ) leads to the reduction of convergence rate and the smaller misadjustment error value. Contrarily, for the bigger ( $\mu$ ) the faster convergence rate is obtained with larger misadjustment error value. So, the selection of the appropriate ( $\mu$ ) is a trade-off between the convergence rate and the misadjustment error and depends on the application that used for. Therefore, the main challenge of this algorithm is the selection of appropriate value for the step size ( $\mu$ ) that guarantees stability [24]. It is obvious from Figure 8 that the tap-weights of the filter are symmetrical around tap-delay number 6.

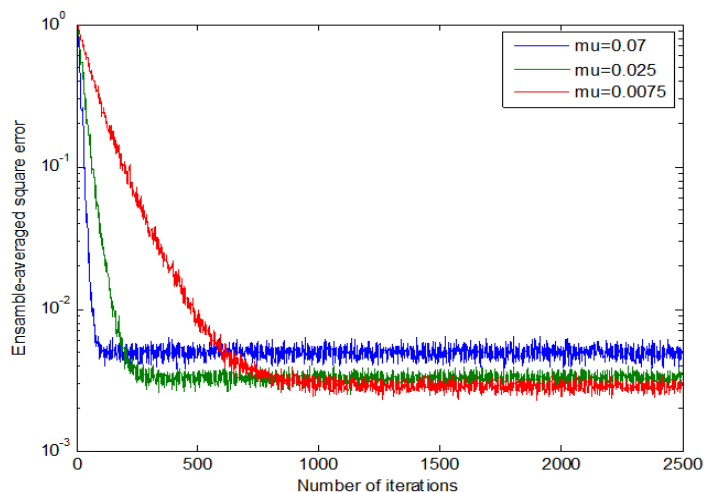


Figure-7. Curves of the LMS algorithm for a filter with  $M=11$ ,  $w=2.9$ , and varying  $\mu$ .

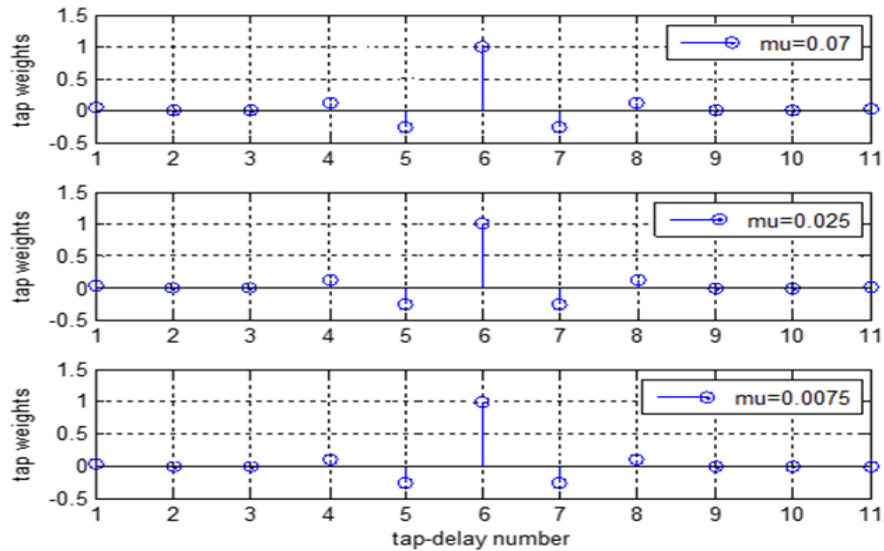


Figure-8. The equalizer impulse response for single channel with various step-size  $\mu$  using LMS algorithm.

3.4. Effect of Decision Feedback Equalizer (DFE)

The inter symbol interference (ISI) significantly slows down the rate of transmitting data in digital communication. This phenomenon is generated by the impact of neighboring symbols on the current symbol. To tackle this problem, we propose the decision feedback equalizer (DFE) technique. This method utilizes the old decisions to improve the system performance.

Figure 9 illustrates the block diagram of DFE which comprises two filters. First filter is feed forward ATF that has  $u(n)$  as input data and second filter is feedback ATF that has the previous decision  $\{d(n)\}$  as an input. The main purpose of the feedback ATF is to filter out the ISI that is generated by previously detected symbols from the predicted symbols [32]. The following equations describe this methodology:

Let us consider  $w_1$ = weighting vector for feed-forward ATF, and  $w_2$ =weighting vector for feedback of ATF, then

$$y(n) = w_1^T u(n); x(n)=w_2^T d(n)$$

$$d'(n)=y(n)-x(n)=[w_1^T - w_2^T] [u(n) \ d(n)]^T$$

$$e(n) = d(n) - d'(n)$$

where  $d(n)$  represents the reference data which is equal to  $\{a_n\}$  delayed by 6 samples.

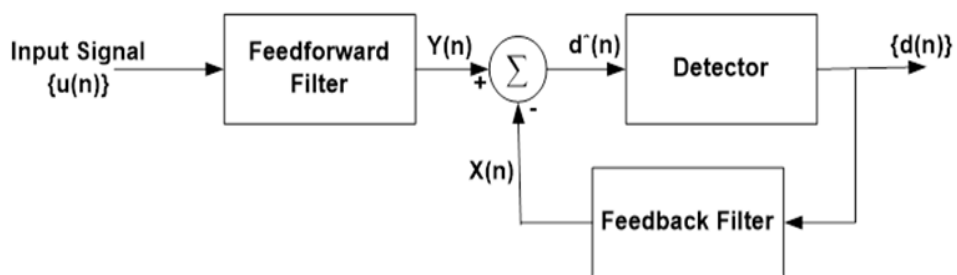


Figure-9. Block diagram of a DFE.

Figure 10 demonstrates the learning curves of the DFE for the channels corresponding to  $w=3.3$ , and  $3.5$ . In both cases, we take the forward ATF order  $M_1= 11$ , feedback ATF order  $M_2 =3$ , and using the same step-size parameter  $\mu = 0.07$ . Comparing both learning curves of the two channels, it is clear that this method shows less sensitivity to eigenvalues spread than the cases without feedback.

In terms of convergence rate, the DFE shows higher convergence speed for both cases and the better equalizer performance so that the MSE is reduced more than 40 times compared to the case without feedback



( $w=3.3$ ) as shown in Figure 11 below. Also Figure 12 shows the tap weights for the DFE for both forward and feedback ATF after averaging 200 independent runs at last iteration (2500 samples). Unlike the other researches conducted before, regarding the same problem, the result obtained with this technique demonstrates its novelty because of such significant reduction of the MSE and consequently the reduction of noise in the system.

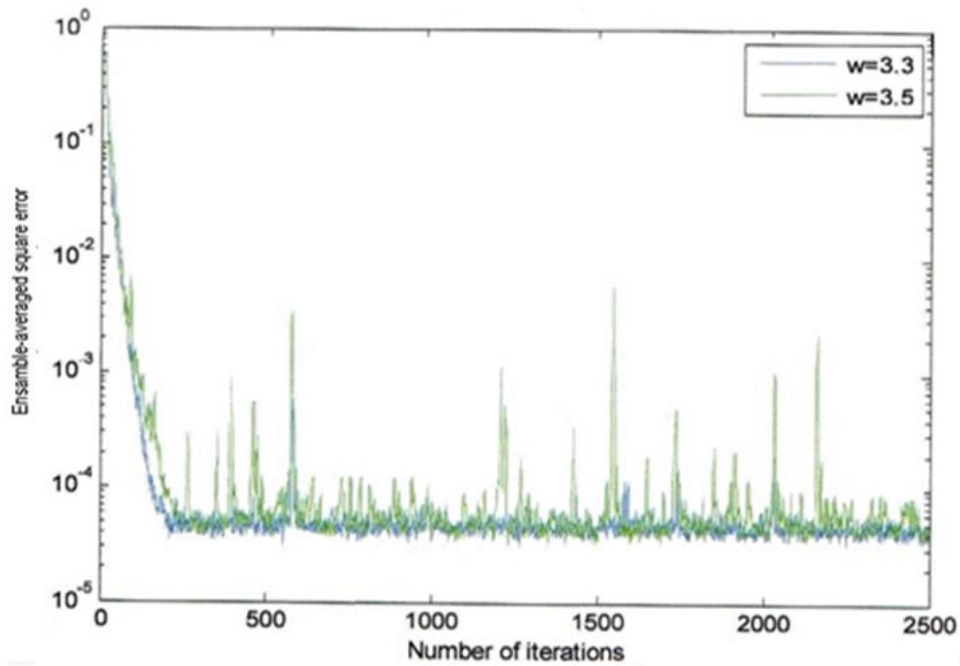


Figure-10. Curves of LMS algorithm of DFE for two different channels with fixed  $\mu$ .

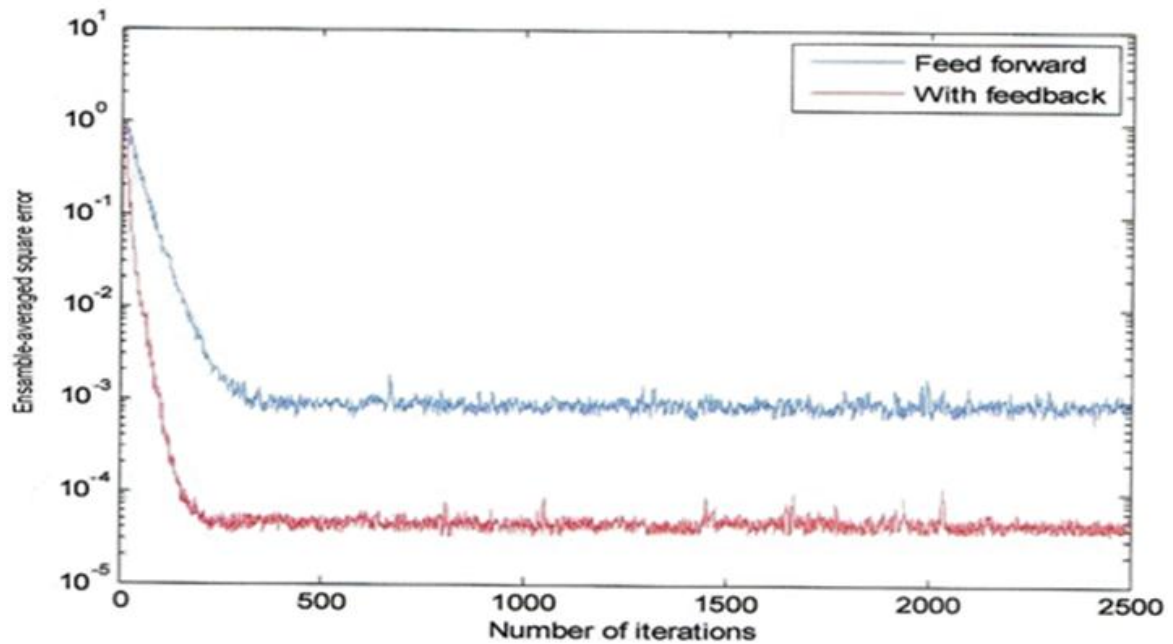


Figure-11. Comparison of the curves of the LMS algorithm of adaptive equalizer with and without feedback of fixed  $\mu$  and  $w=3.3$ .

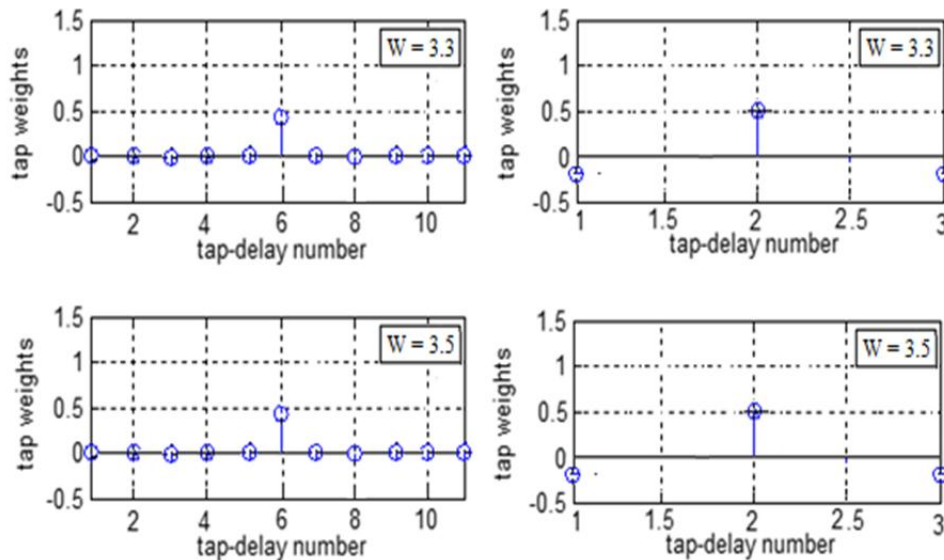


Figure-12. Impulse response of DFE for two different channels using LMS algorithm.

#### 4. COMPARISON BETWEEN LMS ALGORITHM AND RLS ALGORITHM

It is very necessary to compare this algorithm with others in the field of adaptive filter such as RLS algorithm in order to check the performance of this strategy in terms of their convergence rate of speed, sensitivity to channel distortion, the MSE, decision feedback equalizer and computational complexity. Here, we explain the prominent differences between both algorithms throughout their application on one problem.

##### 4.1. Rate of Convergence

- LMS: This algorithm convergence speed is very sensitive to the eigenvalues spread variations. It is much slower than the RLS algorithm. As shown in Figure 5, the LMS technique converges to the steady state MSE after 160 iterations (for  $w = 2.9$ ) and after about 500 iterations for ( $w = 3.5$ ).
- RLS: The speed of convergence of this strategy is relatively insensitive to the eigenvalue spreads variations. The RLS converges about 20 iterations [33].

##### 4.2. Ensemble-Averaged Square Error

- LMS: The averaged MSE in case of this algorithm is more sensitive than the RLS algorithm. As shown in Figure 5, the range of variations in averaged MSE is from 0.004 for  $w = 3.5$  to 0.0004 for  $w = 2.9$ .
- RLS: The averaged MSE in this case is less sensitive to eigenvalues spread ( $w = 2.9$ ) and SNR of 40 dB.

##### 4.3. Computational Complexity

- LMS: This algorithm has the lowest computational complicity a among all algorithms in the field of adaptive filters.
- RLS: This is much more complicated than LMS algorithm in computation complexity and implementation. More over LMS algorithm has a lower SNR compared with RLS technique [34].

#### 5. SUMMARY AND CONCLUSION

We have profoundly discussed the results obtained using LMS algorithm in our study. It is clear that the strategy of cost-effective design of the system has been applied through the selection of the lower filter order (11) for the algorithm better performance. The impacts of eigenvalue spread and step-size parameter ( $\mu$ ) on LMS algorithm performance in terms of MSE reduction and convergence rate for different values of  $w$  and ( $\mu$ ) have been comprehensively analyzed. Moreover, the equalizer impulse response of different channels for eigenvalue spread

and single channel with different ( $\mu$ ) for step-size parameter effect have been plotted and discussed. We conclude that the channels with lower  $w$  have a better performance in terms of both MSE and convergence for fixed step-size parameter ( $\mu$ ). On the other hand, the choice of step-size parameter ( $\mu$ ) for a fixed eigenvalue is a trade-off between convergence rate and misadjustment error and depends on the application that used for.

Unlike the other works performed in LMS technique, we have selected the decision feed-back equalizer (DFE) to improve the system efficiency and convergence rate. Comparison of learning curves of LMS algorithm of adaptive equalizer with and without feedback of fixed  $\mu$  and same eigenvalue spread, shows higher convergence speed and better equalizer performance in case of employing feed-back filter so that the averaged MSE has been reduced more than 40 times with feed-back filter. Therefore, we recommend this decision feedback equalizer for other algorithms in the field of noise cancellation in communication channels.

**Funding:** This study received no specific financial support.

**Competing Interests:** The author declares that there are no conflicts of interests regarding the publication of this paper.

## REFERENCES

- [1] M. Vrankic, D. Sersic, and V. Sucic, "Adaptive 2-D wavelet transform based on the lifting scheme with preserved vanishing moments," *IEEE Transactions on Image Processing*, vol. 19, pp. 1987-2004, 2010. Available at: <https://doi.org/10.1109/tip.2010.2045688>.
- [2] P. Sathawane and D. Prasanthi, "An optimal low power adaptive filter design for noise reduction," *International Journal of Science, Engineering and Technology Research*, vol. 3, pp. 2405-2410, 2014.
- [3] S.-W. Sohn, Y.-B. Lim, J.-J. Yun, H. Choi, and H.-D. Bae, "A filter bank and a self-tuning adaptive filter for the harmonic and interharmonic estimation in power signals," *IEEE Transactions on Instrumentation and Measurement*, vol. 61, pp. 64-73, 2011. Available at: <https://doi.org/10.1109/tim.2011.2150610>.
- [4] L. Stankovic, "Performance analysis of the adaptive algorithm for bias-to-variance tradeoff," *IEEE Transactions on Signal Processing*, vol. 52, pp. 1228-1234, 2004. Available at: <https://doi.org/10.1109/tsp.2004.826179>.
- [5] J. Cerqueira and S. Haddad, "Design of a low power adaptive LMS equalizer for hearing-aid applications," in *2014 IEEE Biomedical Circuits and Systems Conference (Bio CHS) Proceedings. Lausanne, Switzerland, Oct. 22-24, 2014*.
- [6] Y. Ahmed and A. Hoballah, "Adaptive filter-FLC integration for torque ripples minimization in PMSM using PSO," *International Journal of Power Electronics and Drive Systems*, vol. 10, pp. 48-57, 2019. Available at: <https://doi.org/10.11591/ijpeds.v10.i1.pp48-57>.
- [7] M. S. Salman, A. Eleyan, and B. Al-Sheikh, "Discrete wavelet transform-based RI adaptive algorithm for system identification," *International Journal of Electrical and Computer Engineering*, vol. 10, pp. 2383-2391, 2020. Available at: <https://doi.org/10.11591/ijece.v10i3.pp2383-2391>.
- [8] G. Samanta and A. Chandra, "A novel design strategy of low-pass FIR filter using opposition-based differential evolution algorithm," in *2012 IEEE Students Conference on Electrical, Electronic and Computer Science (SCEECS), Bhopal, India, Mar. 1-2, 2012*, pp. 1-4.
- [9] T. Moon, "Universal switching FIR filtering," *IEEE Transactions on Signal Processing*, vol. 60, pp. 1460-1464, 2011. Available at: <https://doi.org/10.1109/tsp.2011.2176931>.
- [10] K. e. a. Rana, "Efficient FIR filter designs using constrained genetic algorithms based optimization," in *2016 2nd international Conference on Communication Control and Intelligent Systems (CCIS), Muthaura, India, Nov.18-20, 2016*.
- [11] S. K. Saha, S. P. Ghoshal, R. Kar, and D. Mandal, "Cat swarm optimization algorithm for optimal linear phase FIR filter design," *ISA Transactions*, vol. 52, pp. 781-794, 2013. Available at: <https://doi.org/10.1016/j.isatra.2013.07.009>.
- [12] M. Sababha and M. Zohdy, "Linear phase FIR low pass filter design based on firefly algorithm," *International Journal of Electrical and Computer Engineering (IJECE)*, vol. 8, pp. 4356-4365, 2018. Available at: <https://doi.org/10.11591/ijece.v8i6.pp4356-4365>.

- [13] M. Rajmohan and H. Shekhar, "Design of parallel and pipelined DA based OBC FIR filter for software defined radio," *Indonesian Journal of Electrical Engineering and Computer Science*, vol. 14, pp. 1228-1234, 2019. Available at: <https://doi.org/10.11591/ijeecs.v14.i3.pp1228-1234>.
- [14] H. Chen, S. Chan, and K. Ho, "A semi-definite programming (SDP) method for designing," *Proceedings, Vancouver*, vol. 3, pp. 149-152, 2004.
- [15] S. C. e. a. Chan, "A new method for designing FIR/IIR digital correction filters for time-interleaved analog-to-digital converter using second order cone programming," in *IEEE 50th Midwest Symposium on Circuits and Systems, Montreal, Que, Canada*, 2007, pp. 1030 – 1033.
- [16] P. Dighe and M. Chinchamalature, "High speed multiplier as IIR filter design using vedic mathematics," *International Journal for technological research in engineering*, vol. 4, pp. 1198-2201, 2017.
- [17] Q. Liu, Y. C. Lim, and Z. Lin, "Design of pipelined IIR filters using two-stage frequency-response masking technique," *IEEE Transactions on Circuits and Systems II: Express Briefs*, vol. 66, pp. 873-877, 2019.
- [18] E. e. a. Asmae, "Meta-heuristic techniques for optimal design of analog and digital filter," *Indonesian Journal of Electrical Engineering and Computer Science*, vol. 19, p. 669~679, 2020.
- [19] A. Fominyh, "Application of the steepest descent method to solving differential Inclusions with either free or fixed right end," in *2017 IEEE Conference on Constructive Non Smooth Analysis and Related Topics (CNSA), St. Petersburg, Russia, May 22-27*, 2017, pp. 1-4.
- [20] M. Z. A. Bhotto and A. Antoniou, "Robust quasi-Newton adaptive filtering algorithms," *IEEE Transactions on Circuits and Systems II: Express Briefs*, vol. 58, pp. 537-541, 2011.
- [21] Y. Dong and H. Zhao, "A new proportionate normalized least mean square algorithm for high measurement noise," in *2015 IEEE International Conference on Signal Processing, Communications and Computing (ICSPCC), Ningbo, China, Sep. 19-22*, 2015, pp. 1-5.
- [22] S. M. H. Irid, M. H. Hachemi, E. A. Haroun, and M. Hadjila, "Spectrum sensing with VSS-NLMS process in femto/macro-cell environments," *International Journal of Electrical and Computer Engineering*, vol. 8, pp. 5185-5205, 2018. Available at: <https://doi.org/10.11591/ijece.v8i6.pp5185-5194>.
- [23] P. e. a. Zhu, "A new variable step size LMS algorithm for a pplication to underwater acoustic channel equalization," in *2017 IEEE International Conference on Signal Processing, Communication and Computing (ICSPCC), Xiamen, China, Oct. 22-25*, 2017, pp. 1- 4.
- [24] A. e. a. Mohammad, "Improvement of LMS adaptive noise canceller using uniform Poly-phase digital filter bank," *Indonesian Journal of Electrical Engineering and Computer Science*, vol. 17, pp. 1258-1265, 2020.
- [25] Y. e. a. WE, "Adaptive notch filter based on LMS algorithm and its application in ground tilt data processing," in *2012 2nd IEEE International Conference on Consumer Electronics, Communications and Networks (CECNet), Yichang, China, Apr. 21-23*, 2012, pp. 2345 – 2348.
- [26] C. M. Rao, D. B. S. Charles, M. G. Prasad, and S. Principal, "A variation of LMS algorithm for noise cancellation," *International Journal of Advanced Research in Computer and Communication Engineering*, vol. 2, pp. 2838-2843, 2013.
- [27] S. Prasad and S. Patil, "Implementation of LMS algorithm for system identification," in *2016 IEEE International Conference on Signal and Information Processing (ICONSIP), Vishnupuri, India*, 2016, pp. 1-5.
- [28] N. Krishnamoorthy, I. Rajkumar, J. Alexander, and D. Marshiana, "Performance analysis of bio-signal processing in ocean environment using soft computing techniques," *International Journal of Electrical and Computer Engineering*, vol. 10, p. 2944, 2020.
- [29] S. M. Jung, J.-H. Seo, and P. Park, "Variable step-size non-negative normalised least-mean-square-type algorithm," *IET Signal Processing*, vol. 9, pp. 618-622, 2015.
- [30] H. e. a. Li, "A new LMS algorithm with application to fixed Satellite communications," in *IEEE 3rd International Workshop on Advanced Computational Intelligence (IWACI), Suzhou, China, Aug. 25-27*, 2010, pp. 72-75.

- [31] R. e. a. Ramli, "Objective and subjective evaluations of adaptive noise cancellation systems with selectable algorithms for speech intelligibility," *Bulletin of Electrical Engineering and Informatics*, vol. 7, pp. 570-579, 2018.
- [32] M. e. a. Rahman, "Development of decision feedback equalizer using simplified adaptive algorithms," *Journal of Critical Reviews*, vol. 7, pp. 305-309, 2020.
- [33] H. Bierk and M. Alsaedi, "Recursive least squares algorithm for adaptive transversal equalization of linear dispersive communication channel," *Journal of Engineering Science and Technology, School of Engineering, Taylor University*, vol. 14, pp. 1043-1054, 2019.
- [34] S. e. a. Patel, "Comparative study of LMS & RLS algorithms for adaptive filter design with FPGA," *Progress in Science in Engineering Research Journal*, vol. 2, pp. 185-192, 2014.

*Views and opinions expressed in this article are the views and opinions of the author(s), Review of Computer Engineering Research shall not be responsible or answerable for any loss, damage or liability etc. caused in relation to/arising out of the use of the content.*