IJCDS 1570983224

International Journal of Computing and Digital Systems
ISSN (2210-142X)
Int. J. Com. Dig. Sys. #, No.# (Mon-20..)

Improved LMS Performance for System Identification Application

Wasan Mamdoh Abdulatef[1], Zainab Rami Alomari[1] and Mahmod Ahmed Mahmod[1]

[1] *Communications Department, Electronics Engineering College, Ninevah University, Mosul, Iraq*

E-mail address: wasan.abdulatef@uoninevah.edu.iq, zainab.saleh@uoninevah.edu.iq, Mahmod.mahmod@uoninevah.edu.iq

Received ## Mon. 20##, Revised ## Mon. 20##, Accepted ## Mon. 20##, Published ## Mon. 20##

Abstract: The adaptive filtering algorithm of Normalized Least Mean Square (NLMS) is known to be highly efficient in terms of requiring less number of iterations compared to the reference Least Mean Square (LMS) method, at the cost of having increased computational complexity. The performance of the simple method of LMS is found to be highly dependent on the step size which is assigned and fixed at the beginning of iterations. Throughout the literature, only the range of LMS step size that assures stability is usually suggested, while the selection of the most suitable value of the step size within this range is still not thoroughly studied. This work proposes the use of the step size value of the first iteration in NLMS to adjust the step size value in LMS, which is found through the results to be highly effective in approaching NLMS behavior without having to increase computational burden. Relying on this way to specify the LMS step size can provide simplicity, accuracy and high convergence speed, not only for system identification, but also for many other adaptive filtering applications.

Keywords: Adaptive filters, Convergence, LMS, Step size, System identification

1. Introduction

The adaptive filter is widely used as the main part in many statistical applications in signal processing field. When processing signals resulting from operating in an environment with unknown statistics, the application of an adaptive filter will offer a clever solution because it usually provides a significant improvement in performance over the use of fixed filters created by conventional methods [1].

Adaptively processing signals has recently witnessed a great interest and growth. This has been encouraged by the developments in the fields of VLSI circuits as well as microelectronics. This is because those fields have enabled performing tremendous amounts of computations for various applications, including the processing of digital signals [2]-[4].

In the application of system identification, adaptive filters have proved to be highly effective and stable. Other applications of adaptive filters include acoustic echo and noise cancellation, adaptive line enhancement, channel estimation, adaptive channel equalization and in communications, specifically in Pulse Code Modulation (PCM) [2], [3]. Adaptive filters are assumed to be Finite Impulse Response (FIR) filters, due to being simple and widely applicable in adaptive processing, compared to Infinite Impulse Response (IIR) adaptive filters which suffer from complications resulting from being hardly stable and relying on gradient search techniques that are unreliable to some extent [4]. However, in some special situations, IIR adaptive filters have proved to be very useful.

The work on adaptive filters firstly started in the 1950s, where a range of adaptive filtering algorithms and applications started to be discovered and developed [5]. Meanwhile, and specifically in 1959, a simple form of LMS algorithm was first introduced for adaptive filtering by Bernard Widrow and his student Ted Hoff [6], [7].

Basically, LMS is a common iterative algorithm used in system identification application by adaptively modifying the coefficients of the adaptive filter so as to characterize the unknown system. After a number of iterations and modifying the coefficient values of the adaptive filter according to the error signal after each iteration, those coefficients eventually shall resemble the coefficients of the unknown system.

E-mail:author's email

In addition to be able to be easily implemented using software simulation programs such as Matlab, the adaptive method of LMS has been widely accepted by researchers for hardware implementation due to having simple and straightforward structure. However, in order to implement it using hardware, the original LMS algorithm has to be modified because of having a recursive loop in its coefficient updating formula which prohibits it from being simply pipelined.

When talking about adaptive filtering algorithms, it is important to mention the method of Recursive-Least-Square (RLS). RLS is a well-known adaptive filtering algorithm that is efficiently used in the application of system identification [8]. It was first introduced by Plackett in 1950. RLS has proved to be a highly efficient and accurate measure in system identification, with having a disadvantage resembled by having high computational complexity.

The feature of having a fixed step size makes LMS unique in simplicity of computations and implementation, compared to the algorithms followed [2]. Step size is the parameter that controls the amount added to the values of the adaptive filter's coefficients after each iteration. The selection of the step size in LMS has a direct impact on the convergence speed of the algorithm. Furthermore, selecting a value that is larger than a specific limit has the effect of deriving the algorithm to unstable condition.

The low computational complexity of LMS has the drawback of lowing convergence speed, where the number of iterations required to reach steady state is usually relatively high. A famous developed version of the LMS algorithm is called the Normalized Least Mean Square (NLMS), involving the calculation of a normalized step size and updating it after each iteration based on the power of instantaneous input signal [8]. This readjustment of the step size has the benefit of efficiently reducing convergence time, at the cost of increasing the computational complexity of the algorithm [2].

Many variations of NLMS have been proposed in the literature, based on updating step size after each iteration depending on a specific calculation method [10]. This improves convergence speed but with increased computations. Some of the most popular of those methods may include: Modified NLMS [10], Leaky LMS [11], Sign Error and Sign Data LMS [12], [13], Variable step size LMS [14], frequency response shaped LMS [15], Hybrid LMS [16], Absolute Average Error Adjusted Step-Size LMS [2], [17] and other algorithms. The use of evolutionary techniques is also suggested to improve convergence of adaptive filters in the denoising operation of medical signals [18]. This however adds further computational complexity to the application of adaptive filtering.

This paper suggests a method to determine the exact LMS step size value which provides an efficient behavior of this algorithm in terms of the number of iterations, as well as Mean Squared Error (MSE). Results show that through this selection of step size, LMS approaches NLMS algorithm behavior without sacrificing the simplicity and low computational complexity LMS is known to have. The rest of this paper is organized as follows: section 2 provides the background behind adaptive filtering use in system identification, with the description of both LMS and NLMS methods. Section 3 describes the step size selection method proposed in this work. Section 4 includes the implementation and quality measures description. Results are given and discussed in sections 5 and 6, respectively, and finally the paper is concluded in section 7.

2. BACKGROUND METHODOLOGY

A. System Identification using Adaptive Filtering

In order to formulate system identification problem, suppose having an unknown linear system that is required to be identified. This system may be an FIR or an IIR. In this work, FIR is used for modeling the unknown system using symbol N for filter order [3]. By connecting this system and the adaptive filter in parallel, and exciting both of them using an input signal *x(n)*, an error signal is calculated from the difference between the output of the adaptive filter and the output of the unknown FIR system. This configuration is symbolized by the block diagram shown in Fig. 1.

Thus the principle behind adaptive filtering algorithms used in system identification is to repeatedly update the adaptive filter's coefficients depending on the difference between the output of the adaptive filter and the unknown system output. This updating operation is done repeatedly and eventually, the values of the adaptive filter coefficients will be highly close to those of the unknown system, where minimizing the error value is the target of iterations.

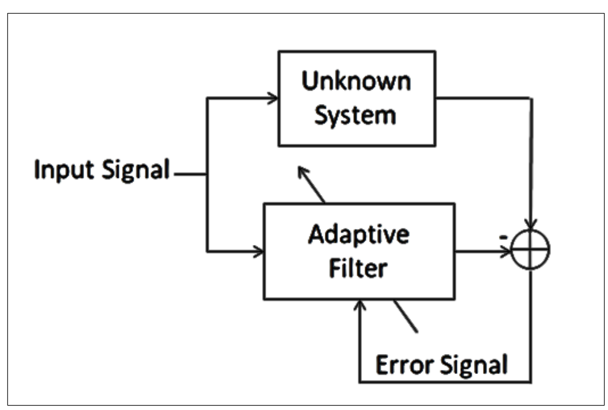


Figure 1. Block diagram of unknown system identification using adaptive filtering.

B. Adaptive Filtering Algorithms

1) LMS Algorithm

Assume to have an FIR filter with an adjustable set of coefficients *W* and an output signal *y(n)*, with the following input signal [2, 3]:

$$x(n) = [x_1(n), x_2(n), ..., x_k(n)], \quad (1)$$

where k is the number of input samples and it is within the range ($0 \le k \le N\text{-}1$), and n is the number of iterations. The filter coefficients are written as follows [2]:

$$W = [W_1, W_2, ..., W_k]. \quad (2)$$

Assume to have a signal y(n) that resembles the output of the adaptive filter, as follows [3]:

$$y(n) = \sum_{k=0}^{N-1} Wx(n-k), \quad n=0, 1, \dots, M \quad (3)$$

In order to use this filter to identify a system with output signal *d(n)* using the configuration shown in Fig. 1, the filter coefficients must be selected so that the output of the filter *y(n)* is comparable to *d(n)* based on this selection. This is called (Adaptive Filter). This adjustment requires the calculation of an error signal *e(n)*. This error signal is calculated as follows [19], [20]:

$$e(n) = d(n) - y(n). \quad (4)$$

The optimum filter coefficients are the coefficients that result in minimizing the squared errors summation. Minimizing *e(n)* depending on the filter coefficients requires differentiating the error signal with respect to each coefficient, yielding a set of linear equations. Solving those equations involves computing the autocorrelation of the input signal in addition to the crosscorrelation between the desired output and the input signal [2], producing a set of optimum coefficient values.

An alternative method that leads to minimizing error signal and yielding optimum filter coefficients without the need for any mathematical evaluation of correlation sequences is the iterative method of LMS. It is known as the simplest and most widely applicable form of adaptive filters. A highly accurate method of RLS is a recursive method used in adaptive filtering. Despite its high accuracy and fast convergence, RLS has the disadvantage of having a high computational complexity, in addition to requiring predefined information and conditions for the coefficient updating operation [21]. This makes RLS method less popular compared to LMS.

In LMS, the filter coefficients are iteratively updated in order to reduce the error signal to the minimum. As the error signal gets lower, the adaptive filter's coefficients become closer to the coefficients of the unknown system [22]. Firstly, a random initial value is assigned to the weights of the adaptive filter. This value can be assumed to be zeros [3]. Error is then calculated based on (1), and then a new value is assigned to the adaptive filter weights according to the following formula [2], [20]:

$$W(n+1) = W(n) + 2\mu.e(n)x(n), \quad (5)$$

where μ is the step size value that is predefined at the start of the algorithm by the user. The selection of the step size is the factor that controls convergence speed of the algorithm towards steady state. Larger values for this step lead to faster convergence, but too large values may drive the algorithm towards unstable behavior and thus the algorithm fails in evaluating the unknown system coefficients. On the other hand, smaller step values lead to slow convergence due to increasing the required number of iterations [3]. Therefore, step size selection must fulfill a trade-off between stability and convergence speed.

LMS has been accused of having a low convergence rate compared to the NLMS algorithm. However, it is going to be shown through the results of this paper that making the right selection of the step size is all that it takes for LMS to produce results that are very near to those obtained using NLMS, with no computations added.

2) NLMS Algorithm

Both the gradient descent methods of LMS and NLMS are based on the continuous updating of the weight values depending on the instantaneous error signal. Unlike LMS, NLMS continuously updates its step size after each iteration. This has the effect of highly improving the convergence rate of NLMS, compared to LMS that uses a single predefined step size.

Step size in NLMS is updated according to the following formula [23]:

$$\mu_{NLMS}(n) = \frac{\delta}{\varepsilon + x_1 x_1^{'}} \quad , \tag{6}$$

where δ is a constant ranging from 1 to 2 and it is usually set to 1 and ε is the smallest non negative value that prevents the case of dividing by zero. Based on that, the values of the weights of the adaptive filters are updated according to the following formula [1]:

$$W(n+1) = W(n) + 2\frac{\delta}{\varepsilon + x_1 x_1^{'}} .e(n)x(n) , \tag{7}$$

3. Proposed Method

As previously explained, the value of the step size highly affects the behavior of LMS algorithm when going towards the optimum solution. Throughout the literature, the limits within which step size must be selected are specified by more than a formula. Table 1 shows a list of references that suggest formulas for specifying the range within which step size in LMS must to be selected to ensure stability. This however leaves a wide range of choices for the user to specify the value of the step from, where a specific value for step size is usually selected within this range without justification [24]-[28]. Table 1 also includes a column showing the value used by a number of references for LMS step size.

TABLE I. THE METHODS AND VALUES OF STEP SIZE RANGE OF LMS ALGORITHM THAT ARE WIDELY USED IN THE LITERATURE FOR SYSTEM IDENTIFICATION APPLICATIONS

Ref.	Method used	Step size value used
[2]	Small +ve value	Not Given
[3]	0< μ <1/10NPx*	Not Given
[24, 25]	0 < μ < (1/ λ_{max}) **	0.0625 and 0.08
[26]	0 < μ < (2/ λ_{max})	from 0.001 to 1
[27]	0 < μ < 1	from 0.001 to 0.1
[28]	0 < μ < 0.2	0.01 and 0.004

* Px denotes the power included in the input signal *x(n)*.
** λ_{max} denotes the maximum eigenvalue of the covariance matrix of the input signal *x(n)*.

It can be noticed that the equations given in table 1 can only be used to specify the limits of the step size, leaving a wide range of feasible values to select the step size from. The optimum step size within a given range has been the material of many studies [26], [27], where LMS behavior is assessed at a range of step sizes and based on the results, the most suitable value is recommended. Nevertheless, this value is vulnerable to change when the filter order, weight values or input signal properties is changed, limiting the efficiency of such studies.

This work suggests adjusting the step size of LMS so that it equals to the step size used in NLMS during the first iteration. This can be described by the following equation:

$$\mu_{LMS} = \mu_{NLMS}(0) = \frac{\delta}{\varepsilon + x_1(0) x_1(0)'} \quad . \tag{8}$$

Unlike step size in NLMS that is continuously updated after each iteration, step size will be fixed throughout the LMS learning process so that no additional computational burden is added.

4. Implementation and Quality Measurements

In order to assess the proposed method, Matlab implementations of the LMS algorithm using various methods for step size selection are produced, together with the NLMS algorithm implementation for comparison.

The input is assumed to be a random signal with a normal distribution, generated using a Matlab function called (*randn*) that generates a sequence of random values drawn from the standard normal distribution, with an amplitude of 100. This signal is generated only once and its values are stored and reused as the input for all the implemented methods.

According to the equations in table 1, maximizing LMS speed of convergence without driving the system to unstable condition is by selecting a step size that equals the maximum limit specified in that equation. Based on this, step sizes are found to be as given in table 2, for filter orders 10 and 20.

A low-pass FIR filter is used to represent the unknown system, using a Matlab function called (*fircband*). This function specifies the filter coefficients based on the given constraints. The desired output signal *d(k)* is found using a Matlab function called (*filter*). *d(k)* will be used for error calculation after each iteration according to (1).

In addition to the plot of the amplitude of the error signal, three different numerical metrics are used to assess the implemented methods. Firstly, the mean squared error in dB scale, measured for the error produced from all iterations. The second performance measure is the Weight Difference (WD) that represents the summation of the absolute difference between unknown system coefficients and the coefficients of the adaptive filter after a specific number of iterations, converted to dB scale. The third measure is the absolute error value resulting after 1000 iterations, in dB scale.

During simulations, the filter order of the adaptive filter is set to be equal to the order of the unknown system. This is not possible in practical cases, where the filter order of the unknown system is usually unknown. However, it is possible to detect the filter order of this system simply as follows: Firstly, a random value for the order of the adaptive filter is selected. Then the result of applying LMS algorithm will give an indication on the filter order of the unknown system as follows:

- If the order selected for the adaptive filter was less than that of the unknown system, the system will either go towards instability condition at the end of the LMS iterations, or high values of error signal will be achieved. In other words, the error will not be minimized at the end of the iterations and this indicates that higher value should be assigned for to adaptive filter order.
- Otherwise if the selected order of the adaptive filter was more than that of the unknown system, LMS algorithm will perform properly with simply giving zeros to the values of the additional weights. This means that the order of the unknown system will be indicated by the number of non-zero coefficients, and the order of adaptive filter will simply be amended to the unknown system's order.

5. RESULTS

The proposed step size selection method is examined through MATLAB implementation of LMS and NLMS algorithms, using the filter orders: 10 and 20. In addition to the proposed method, step size in LMS is selected based on three various well-known methods given in table 1. For a fair comparison, the upper limit in each equation that specifies the range for step size is used, in order to provide the fastest possible convergence to the optimum solution provided by the equation and to assure stability at the same time. Fig. 2 shows the convergence rate in terms of the absolute instantaneous error value for each of the used methods compared to NLMS algorithm, for filter orders 10 and 20.

It can be seen from Fig. 2 that the nearest LMS behavior to NLMS is when using the proposed method for selecting the value of step size, at both of the used filter orders. In addition, subjective results given in table 2 show that it is not only the convergence time that is efficiently reduced compared to the other implemented classic step size selection methods, but the proposed method is also the nearest to NLMS algorithm in each of Mean Square Error (MSE), Weight Difference (WD), and in the value of error signal calculated according to (4) after 1000 iterations. The high match between the actual weights of the unknown system and the weights produced for the adaptive filter using the proposed method after 1000 iterations is shown in Fig. 3, and after getting an error of less than 0.001 (after 366 iterations) is shown in Fig. 4, for filter order of 20.

Another measure of the efficiency of the proposed method is the number of iterations required to start producing an error value of less than 0.001. This measure is given in table 3 for filter orders 10 and 20, where various methods for selecting step size in LMS are implemented for comparison (from table 1), in addition to NLMS.

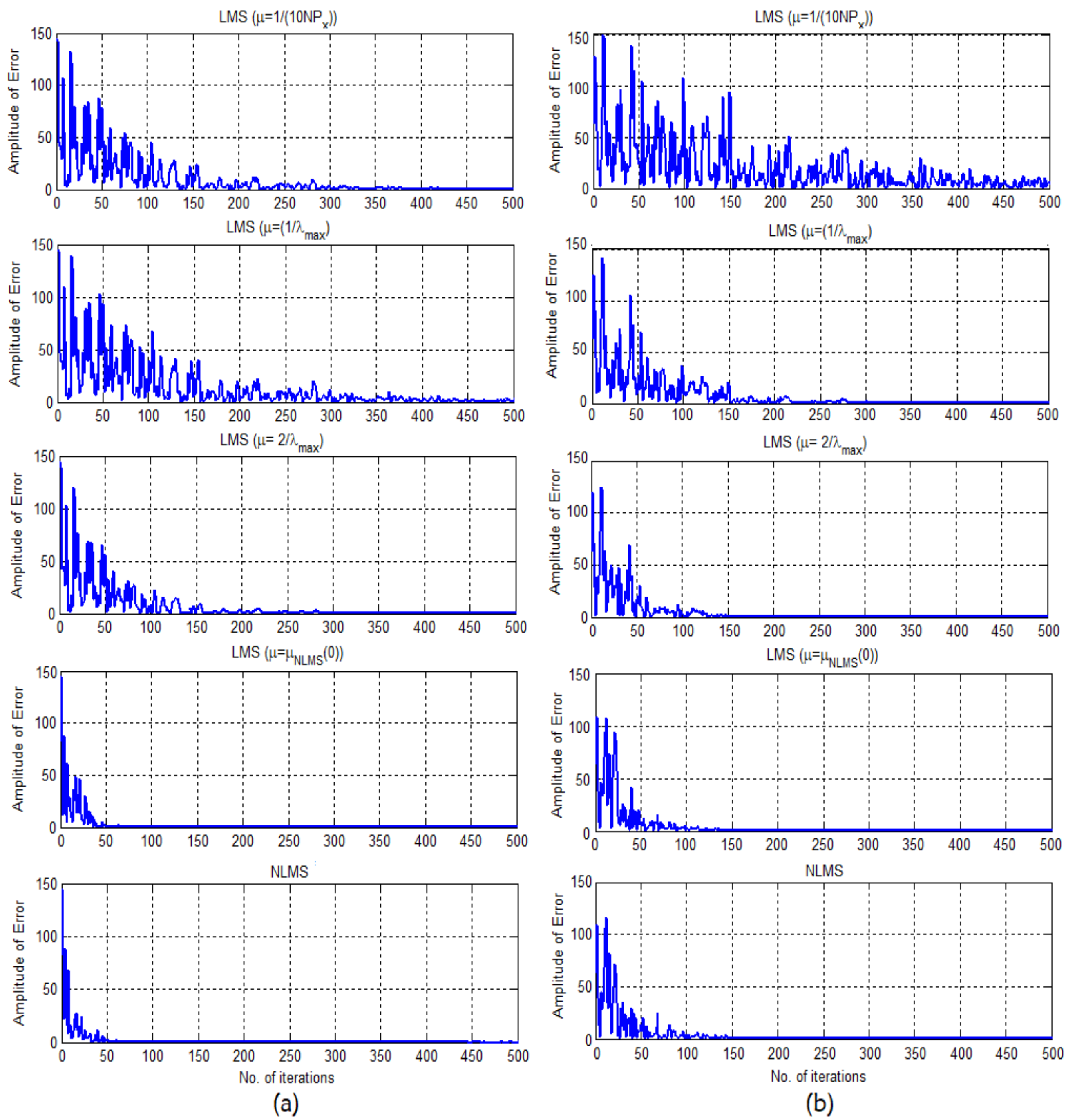


Figure 2. The amplitude of the error achieved by NLMS and by LMS algorithms using various step size selection methods, for filter orders 10 and 20.

TABLE II. PERFORMANCE EVALUATION OF NLMS AND LMS USING VARIOUS STEP SIZE SELECTION METHODS AFTER 1000 ITERATIONS, FOR FILTER ORDERS 10 AND 20

Method	Step Size	MSE at iteration no. 1000 (dB)	WD at iteration no. 1000 (dB)	Error at iteration no. 1000 (dB)
Filter Order = 10				
LMS (μ=1/10NP$_x$)	1.0300e-6	2.3791	-4.4587	-2.7312
LMS (μ=1/ λ_{max})	7.5484e-7	2.5068	-3.2928	-1.7564
LMS (μ=2/ λ_{max})	1.5097e-6	2.2262	-6.4913	-4.6869
LMS (μ= μ_{NLMS}(0))	1.0859e-5	1.7372	-15.3262	-13.8474
NLMS		1.6744	-15.4427	-14.1484
Filter Order = 20				
LMS (μ=1/10NP$_x$)	5.0578e-7	2.6672	-1.7628	-3.1541
LMS (μ=1/ λ_{max})	1.4607e-6	2.2561	-5.6064	-4.6684
LMS (μ=2/ λ_{max})	2.9214e-6	2.0480	-11.1128	-9.8142
LMS (μ= μ_{NLMS}(0))	5.6338e-6	2.0068	-13.1418	-11.3798
NLMS		1.9763	-14.3368	-13.5463

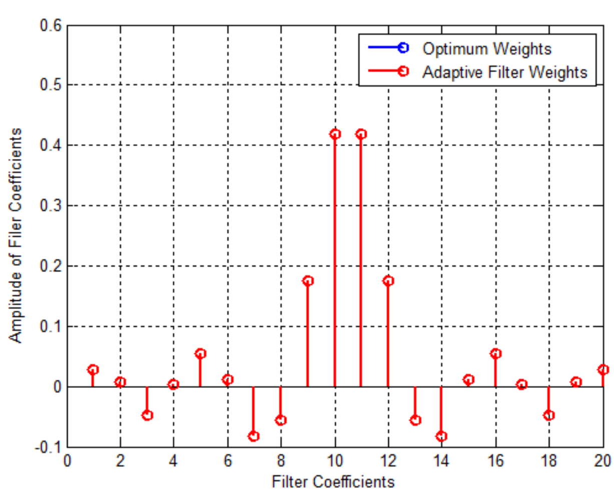


Figure 2. A high match between the optimum weights of the unknown system and the adaptive filter weights after 1000 iterations using the proposed method using a filter order of 20.

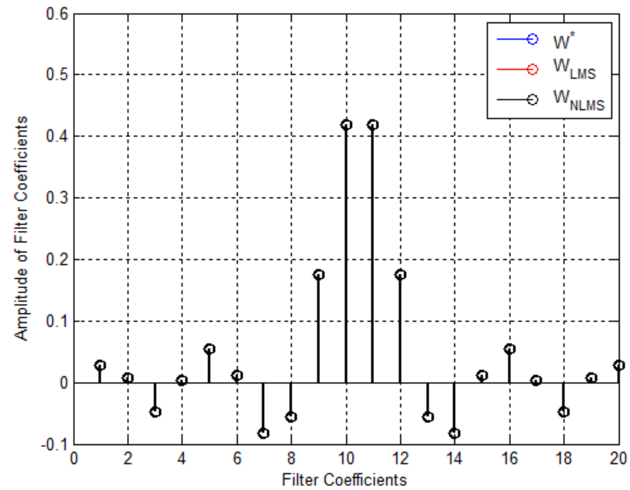


Figure 3. A high match between the actual values of the unknown system weights (W*) and LMS weights using the proposed method after 366 iterations, together with NLMS weights after 320 iterations using a filter order of 20.

In Fig. 5, comparison is carried out using bar representation for filter orders 5, 10, 15, 20 and 25, using the proposed step size selection method as well as the three other step size selection methods, in addition to NLMS method. From the comparison in this figure, it can be clearly noticed that the number of iterations taken by LMS using the proposed method is the nearest to the number of iterations required by NLMS, compared to the other implemented methods.

It is important to mention that the proposed method works well with filter orders of more than 5, where it is found that a second update of step size is required by the proposed method to approach NLMS behavior for filters of order 5 and less. This can be done using the same equation used for creating the value of the first step, which is given in (1). However, this is usually not a problem since FIR filters often use filter orders of more than that. This is a disadvantage of FIR filters over IIR filters, where to achieve a specific level of performance, FIR filters often require a much higher filter order than IIR filters. From one side, during software implementation of FIR filters, increasing filter order practically has no effect on implementation cost. On the other side, a disadvantage of FIR filters that is accompanied with increasing their orders will occur. This disadvantage is resembled by requiring a higher delay compared to an IIR filter with equal performance [29], [30], due to approaching the ideal filter response.

TABLE III. THE NUMBER OF ITERATIONS TAKEN BY THE IMPLEMENTED ADAPTIVE FILTER METHODS TO START PRODUCING THE AN ERROR OF LESS THAN 0.001. THE USED STEP SIZES ARE THE UPPER LIMITS OF THE RANGES GIVEN IN TABLE 2.

Method	No. of Iterations	
	Filter Order=10	Filter Order=20
LMS (μ=1/10NP_x)	>1000	>1000
LMS (μ=1/ λ_{max})	>1000	907
LMS (μ=2/ λ_{max})	766	439
LMS (μ= μ_{NLMS}(0))	219	366
NLMS	157	320

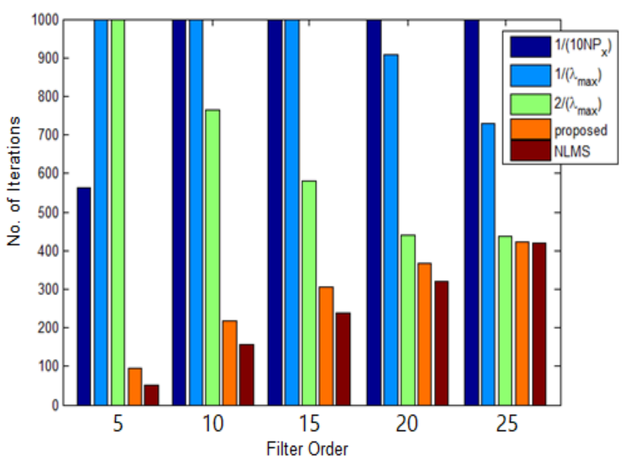


Figure 4. Bar representation of the number of iterations required by various methods used for step size calculation, to achieve an error of less than 0.001, using five different filter orders within the range from 5 to 25.

6. DISCUSSION

The measurements of the number of iterations required by various step size selection methods to reach an error of less than 0.001 show that those methods respond in various ways to increasing the filter order. This can be clearly seen in figure 5. Both of NLMS and the proposed method have a close similar behavior of requiring more iterations as filter order increases. Methods that depend on the inverse of the maximum eigenvalue and the inverse of the signal power have numbers of iterations that are inversely proportional to the filter order. However, the behavior of NLMS and the proposed method are still shown to be superior for the tested range of filter orders, in addition to being so close to each other.

The results given in section 5 of this work successfully confirm that even very small changes in the value assigned to step size in LMS can lead to a huge variation in behavior, in terms of the convergence speed and the amount of instantaneous error in weight estimation. The proposed step size selection method is evaluated at various filter orders for feasibility verification. This is to prove that making the right selection of step size highly affects the behavior of LMS algorithm and furthermore, it leads to approaching the efficient behavior of the algorithm of NLMS.

As shown in tables 2 and 3, the performance of LMS algorithm is highly sensitive even to very small amounts of changes in step size. According to these tables, a change of (1.4607e-06) in step size for a filter of order 20 results in changing the number of iterations required to reach an error of 0.001 to nearly the half (from 907 to 439 iterations). Thus it is not feasible to specify a range that is relatively wide for step size without ruling the selection of the step within that range.

Improving the performance of adaptive filters that are used for system identification can be as simple as selecting the correct step size, with no need to update the step size with each iteration. This helps improve the performance without sacrificing the computational simplicity. The proposed method of step size specification shows a superior behavior, which is confirmed through the measurements of the number of occupied iterations required to produce 0.001 error and in both MSE and Weight Difference (WD) measurements.

This work is dedicated to prove the efficiency of the method proposed for selecting step size in LMS for the application of system identification that is widely used in many recent systems. This method is not tested for other applications of adaptive filtering yet. However, it is expected to have a similar behavior in approaching NLMS performance. This is planned to be performed in future works, where the proposed method will be tested for the application of noise removal in communication systems and signals. In addition, the range around the values specified by the proposed method for LMS step size are planned to be studied, in order to select an ultimate range that represent the optimal solution for step size selection in for the general method of LMS that can be used in any application.

7. CONCLUSION

It is found in this work that the well-known iterative method of LMS is able produce a superior behavior by simply setting its step size to that used during the first iteration of NLMS algorithm. This ensures no more computations are added to the original LMS. The results found through this work using Matlab implementation show the efficiency of the proposed step size selection method and thus clarify the importance of the step size in directing the whole behavior of LMS. Further investigations over the range that is around the values of the step size specified in this research may lead to further improvement of LMS algorithm performance. Furthermore, the proposed method can be proved to be effective not only for the application of system identification, but also for other applications of adaptive filters. This takes us to the next phase of this research that is planned to involve testing the suggested step size selection method for the application of noise cancellation.

ACKNOWLEDGMENT

Authors would like to thank the Ministry of Higher Education and Scientific Research of Iraq, as well as the Electronics Engineering College, Ninevah University, for their continuous encouragement and support of scientific research. Authors are also thankful to the reviewers and editorial team of the journal of IJCDS for their valuable reviews and efforts in publishing this work.

REFERENCES

[1] S. Dixit and D. Nagaria, LMS adaptive filters for noise cancellation: A review. International Journal of Electrical and Computer Engineering (IJECE), Vol. 7, no. 5, 2017, pp. 2520-2529.

[2] G.S. Nariman, H. D. Majeed, Adaptive filter based on absolute average error adaptive algorithm for modeling System, UHD Journal of Science and Technology, Vol. 6, No. 1, May 2022, pp. 60 – 69.

[3] V. K. Ingle and J.G. Proakis, Digital signal processing using Matlab: a problem solving companion, 3rd Ed., Cengage Learning, 2012.

[4] S. D. Stearns, Fundamentals of adaptive signal processing, Sandia national laboratories, 1985.

[5] W. Liu, J. C. Principe and S. Haykin, Kernel adaptive filtering: a comprehensive introduction, John Wiley & Sons Inc., 2010.

[6] B. Widrow, Thinking about thinking: the discovery of the LMS algorithm, IEEE Signal Processing Magazine, Vol. 22, No. 1, January 2005, pp. 100 – 106.

[7] B. Widrow, Recollections of Norbert Wiener and the first IFAC world congress, IEEE Control Systems Magazine, Vol. 21, No. 3, June 2001, pp. 65 – 70.

[8] S. Haykin, Adaptive filter theory, 5th Ed., Pearson Education India, 2002.

[9] P. E. An, B. Brown and C. J. Harris, On the convergence rate performance of the normalized Least-Mean-Square adaptation, IEEE Transactions on Neural Networks, Vol. 8, No. 5, September 1997, pp. 1211 – 1214.

[10] T. Chulajata, H. M. Kwon and K. Y. Min, Adaptive antenna array with no phase calibration for CDMA reverse link, 52nd Vehicular Technology Conference, September 24-28, 2000, Boston, MA, USA.

[11] A. D. Poularikas and Z. M. Ramadan, Adaptive filtering primer with Matlab, CRC Press, 2006.

[12] N. L. Freire and S. C. Douglas, Adaptive cancellation of geomagnetic background noise using a Sign-Error normalized LMS algorithm, IEEE International Conference on Acoustics, Speech, and Signal Processing, 06 August 2002, Minneapolis, MN, USA.

[13] B. Paul and P. Mythili, ECG noise removal using GA tuned sign-data Least Mean Square algorithm, IEEE International Conference on Advanced Communication Control and Computing Technologies (ICACCCT), 04 October 2012, Ramanathapuram, India, pp. 100 – 103.

[14] K. W. Chan and Y. T. Zhang, Adaptive reduction of motion artifact from photoplethysmographic recordings using a variable step-size LMS filter, Sensors, June 12-14, 2002, pp. 1343 – 1346, Orlando, FL, USA.

[15] O. Kukrer and A. Hocanin, Frequency-response-shaped LMS adaptive filter, Digital Signal Processing, Vol. 16, No. 6, November 2006, pp. 855 – 69.

[16] P. C. Chang, C. S. Yu and T. H. Lee, Hybrid LMS-MMSE inverse halftoning technique, IEEE Transactions on Image Processing, Vol. 10, No. 1, January 2001, pp. 95 – 103.

[17] T. M. Jamel and H. A. Mohamed, Noise canceller using a new modified adaptive step size LMS algorithm, WSEAS Transaction on signal processing, Vol. 10, August 2014, pp. 637 – 44.

[18] M. K. Ahirwal, A. Kumar and G. K. Singh, Adaptive filtering of EEG/ERP through noise cancellers using an improved PSO algorithm, Swarm and Evolutionary Computation, Vol. 14, February 2014, pp. 76 – 91.

[19] S. Kalluri and G. R. Arce, A general class of nonlinear normalized adaptive filtering algorithms, IEEE Transactions on Signal Processing, Vol. 47, No. 8, August 1999, pp.2262 – 2272.

[20] A. Ali, M. Moinuddin and T. Y. Al-Naffouri, NLMS is more robust to input-correlation than LMS: A proof, IEEE Signal Processing Letters, Vol. 29,December 2021, pp. 279 – 283.

[21] S. K. Kiran and N. K. Mahadevaswamy, System identification of FIR filters, Journal of Engineering Research and Sciences, Vol. 1, No. 4, April 2022, pp. 74 – 80.

[22] The Math Works, Inc., MATLAB, version R2023b, accessed March 10, 2024, https://www.mathworks.com/help/dsp/ug/system-identification-fir-filter-using-lms-algorithm.html.

[23] S. Ciochina, C. Paleologu and J. Benesty, An optimized NLMS algorithm for system identification, Signal Processing, Vol. 118, January 2016, pp. 115 – 121.

[24] M. Santhi and C. T. Nallammai, Low complexity implementation of LMS/ZALMS adaptive filter for sparse system identification, Middle-East Journal of Scientific Research, Vol. 24, No. 2, 2016, pp. 120 – 126.

[25] S.D. Shinde, M.A. Patil and M. P. Belagali, System identification using adaptive filter algorithms, IOSR journal of electronics and communication engineering, pp. 54-57.

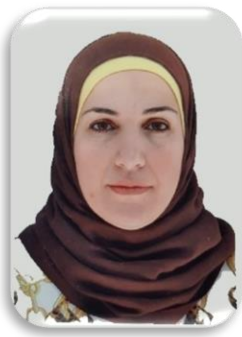
[26] S. R. Prasad and B. B. Godbole, Optimization of LMS algorithm for system identification, arXiv preprint arXiv:1706.00897, 2017.

[27] F. R. Jiménez-López, C. E. Pardo-Beainy and E. A. Gutiérrez-Cáceres, Adaptive filtering implemented over TMS320c6713 DSP platform for system identification, Iteckne, Vol. 11, No. 2, December 2014, pp. 157 – 171.

[28] K. Rasadia and K. Parmar, Adaptive filter analysis for system identification using various adaptive algorithm, International Journal of Engineering Research and Technology (IJERT), Vol. 1, No. 3, May 2012.

[29] The Math Works, Inc., MATLAB, version R2023b, accessed March 19, 2024, https://www.mathworks.com/help/signal/ug/fir-filter-design.html.

[30] Z. Milivojevic, "Digital filter design", MikroElektronika, 2009.



Wasan M. Abdulatef is a faculty in the Department of Communications at Electronics Engineering College, Ninevah University. Her research interests are focused on MIMO and OFDM in wireless communications and signal processing. She taught several B.Sc. modules such as signals and systems, Satellites, Communications Electronics, Digital Signal Processing, mathematics, DSP and Matlab labs and other subjects. In the recent time, she is pursuing her Ph.D. studies in the department of electrical engineering in Mosul University.

Dr. Zainab Alomari is a faculty in the Department of Communications at Electronics Engineering College, Ninevah University. Both of her B.Sc and M.Sc degrees were in computer engineering from Mosul University. She received her Ph.D. from Leeds University, by working with the ultrasound group on ultrasound imaging beamforming techniques. Recently she is teaching Microprocessors, DSP, Matlab Labs and other modules for B.Sc. and M.Sc. studies, and supervising undergraduate as well as postgraduate projects.

Dr. Mahmod A. Mahmod is an assistant Professor, who is recently the head of Communications Department in Electronics Engineering College, Ninevah University. He has obtained his B. Sc., M. Sc. and Ph.D. from the department of Electrical Engineering, Mosul University, in the specialty of Electronics and Communications Engineering. He teaches various B.Sc. and M.Sc. modules such as Digital Communications, Radar Communications and Digital Signal Processing, in addition to supervising undergraduate and postgraduate projects.