Development and Comparative Study of Least Mean Square-Based Adaptive Filter Algorithms

Akpan, Nsikak-Abasi Peter[1] , Udofia, Kufre[2] , Ozuomba, Simeon[3]
[1]Department of Electrical/Electronic Engineering, Akwa Ibom State University Mkpat Enin, Akwa Ibom State, Nigeria.
[2,3]*Department of Electrical/Electronic and Computer Engineering, University of Uyo, AkwaIbom, Nigeria.*
([3]simeonoz@yahoo.com)

***Abstract*— In this paper five different Least Mean Square (LMS)-based adaptive filter algorithms are presented. Then, a modified version of the LMS algorithm is proposed which combines the step size adaptation mechanisms of the "pure" LMS algorithm and that of the Normalized Least Mean Square (NLMS) algorithm. The performance of the various LMS-based adaptive filter algorithms presented in this paper are then evaluated and compared based on Mathlab simulation of noise cancellation using each of the LMS algorithms. In the simulation, the input signal (or desired signal) is from Hallelujah chorus by Handel whereas Gaussian noise was used as noise signal. The mixture of the input signal and noise signal was subjected to filtering using each of the five adaptive filter algorithms presented in this paper as well as the proposed adaptive filter algorithm developed in this paper. The plot of the output signal, the error signal, the combination of desired signal, output signal and error signal and the coefficient output are given for each algorithm along with the R-square values, sum of square error and root mean square error of different models. The results show that the proposed modified LMS model is the best model for cancelling noise in the desired signal (Handel's Hallelujah chorus) since it has the highest R-square value of 99.83 %, the lowest SSE value of 5.0011 and the lowest RMSE value of 0.0081. The next model in performance rating is the NLMS with R-square value of 99.81 %, SSE value of 5.0112 and RMSE value of 0.0083. On the other hand, the sign-sign least mean square (SSLMS) model has the highest sum of square error value of 99.9945. This implies that it is inefficient to use SSLMS to cancel noise in the given case considered in this paper.**

Keywords—Adaptive Filter, Normalized Least Mean Square (NLMS) , Sign-Sign Least Mean Square (SSLMS), Sign-Data Least Mean Square Model (SDLMS), Sign-Sign Least Mean Square (SSLMS), Sign-Error Least Mean Square (SELSM)

I. INTRODUCTION

In many digital signal processing applications such as channel equalization, echo cancellation and noise cancellation the second order statistics cannot be effectively specified [1, 2, 3, 4,5]. In such applications, adaptive filters with adjustable coefficients are employed. Basically, adaptive filter automatically adjusts its transfer function according to its operating optimizing algorithm that allows the filter coefficients to adapt to the signal statics [2, 3, 5, 6, 7]. Essentially, the adaptive filter adapts its performance based on the input signal; the algorithm enables it to adjust its parameters to produce an output that matches the output of an unknown system [2, 8, 9, 10,11].

There are different approaches used in adaptive filtering. However, in those applications where adaptive filtering is needed, the Least Mean Squares (LMS) algorithm is the most widely used algorithm [12, 13, 14, 15, 16, 17, 18, 19]. Particularly, compared to other adaptive filtering techniques, LMS algorithm has been found to be simple, fast, and robust [20, 21, 22, 23]. Importantly, the algorithm step size is the key intrinsic feature of the LMS algorithm and the step size requires careful adjustment. In order to realize small excess mean square error small step size may be used but it results in slow convergence. On the other hand, using large step size leads to fast adaptation and hence fast convergence, but it may result in loss of stability. Essentially, the main drawback of the "pure" LMS algorithm is that it is sensitive to the step size . This makes it very hard to choose a learning rate that ensures stability of the algorithm [24]. The Normalised least mean squares filter (NLMS) is a variant of the LMS algorithm that solves this problem by normalising with the power of the input [2, 25]. Accordingly, researchers have continued to develop many version of the LMS algorithm with different step size adaptation mechanisms.

In this paper some LMS algorithms are presented and a modified version of the LMS algorithm is proposed which combines the step size adaptation mechanisms of the "pure" LMS algorithm and that of the NLMS algorithm. The performance of the various LMS algorithms presented in this paper are then evaluated and compared based on Mathlab simulation of noise cancellation using each of the LMS algorithms.

II. LITERATURE REVIEW

An adaptive filter is a time variant filter whose coefficients are adjusted in a way to optimize the error of the signal or to satisfy some predetermined optimization criterion [25, 26, 27]. They can automatically adapt (self-optimize) in the face of changing system requirements and they can be trained to perform specific filtering and decision making tasks according to some updating equations. In general, any system with a finite number of parameters that affect how y(n) is computed from x(n) could be used for the adaptive filter. Adaptive filter finds application in many signal processing problems , however, in this paper, the focus is on noise cancellation. Figure 1 shows a simplified noise cancellation via adaptive filter.

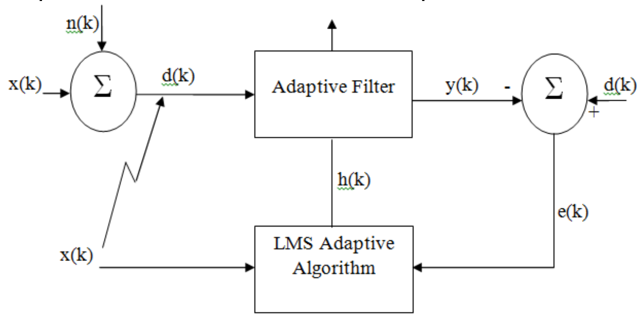

Figure 1: Simplified noise cancellation via adaptive filter

Based on figure 1, x(k) is the input signal, n(k) is the noise, d(k) is the desired response, h(k) is the impulse response of adaptive filter, y(k) is the filtered output and e(k) is the error signal. The objective function (which is minimizing the error) is given by:

$$\sum_{k=0}^{N-1} e^2(k) \ or \ E\left(e^k(k)\right) \quad (1)$$

where: $e^2(k)$ is the estimation error, N is the number of iteration and E is the expectation operation.

There different types of via adaptive filters used for noise cancellation. However, the Least Mean Square (LSM) method has proven to be the most popular. Accordingly, over the years various types of LMS adaptive filter have been developed. Some of these LMS adaptive filter are considered in this section and is the next section of this paper, a proposed modified version of some existing LMS adaptive filter is presented and its performance is compared with those of the existing LMS adaptive filter presented in the paper.

A. Least Mean Square Adaptive Filter

The flowchart for the LMS adaptive filter is shown in Figure 2.

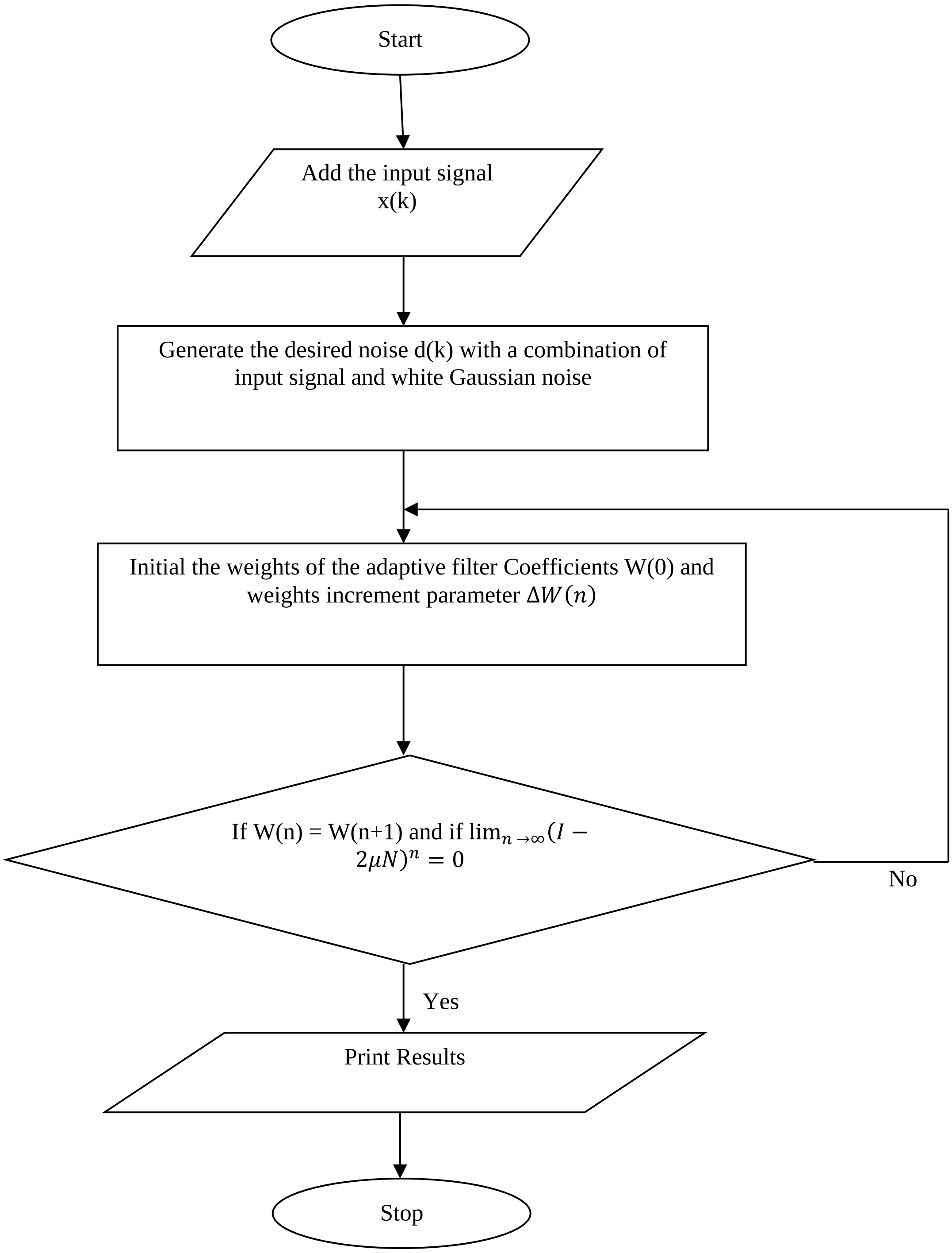

Figure 2: Flowchart for the LMS algorithm

The least mean square model was carried out with steepest descent method of optimization [7, 11, 28, 29, 30]. The objective function of LMS is to cancel out noise in the desired signal which is of the form:

$$E(e^2(k)) \quad (2)$$

where: E is the expectation operation, $e(k)$ is the estimation error at time n given by

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

Where:

$$y(n) = \sum_{i=0}^{L-1} w_i(n)x(n-i) \quad (3)$$

$$W(n) = [w_0(n)\ w_1(n) \ldots w_{L-1}(n)\ w_{L-2}(n)]^T \quad (5)$$

$$X(n) = [x(n)\ x(n-1) \ldots x(n-L+2)\ x(n-L+1)]^T \quad (6)$$

The expected solution after filtering was:

$$W(n+1) = W(n) + \Delta W(n) \quad (7)$$

where: $\Delta W(n)$ is an incrementing factor, n is the time, W(n) is the total weights to be adjusted at time n, X(n) is the total input signal at time n, w(n) and x(n) are the weights and input signals of each adaptive filter coefficients respectively. The steepest descent gradient searching was used.

The steepest descent is stable and will converge if:

$$\lim_{n\to\infty}(I - 2\mu N)^n = 0 \quad (8)$$

Where μ is the step size (or convergence factor) that determines the stability and the convergence rate of the algorithm.

B. Normalized Least Mean Square Adaptive Filter Model

The normalized least mean square filter is a variant of least mean square filter that normalizes the power of the input [13, 18, 31]. The model can be summarized as follows;

$$x(n) = [x(n), x(n-1), \dots, x(n-L+1)]^T \tag{9}$$

$$e(n) = d(n) - h^H(n)x(n) \tag{10}$$

$$h(n+1) = h(n) + \left(\frac{\mu e^*(n)x(n)}{x^H(n)x(n)}\right) \tag{11}$$

where: x(n) is the input signal, μ is the learning rate that guarantees the stability of the algorithm, L is the filter order, e(n) is the error of the current sample n, d(n) is the desired signal of the current sample n, h(n) is the impulse response, H or T is the conjugate transpose. If there is no interference (v(n) = 0), the optimal learning rate for the NLMS algorithm is;

$$\mu_{opt} = 1 \tag{12}$$

In other words the learning rate is independent of the input x(n) and real impulse response h(n). But since there is interference, the optimal learning rate is;

$$\mu_{opt} = \frac{E\left[|y(n)-y^*(n)|^2\right]}{E[|e(n)|^2]} \tag{13}$$

The flowchart for the normalized least mean square (LMS) adaptive filter is shown in Figure 3.

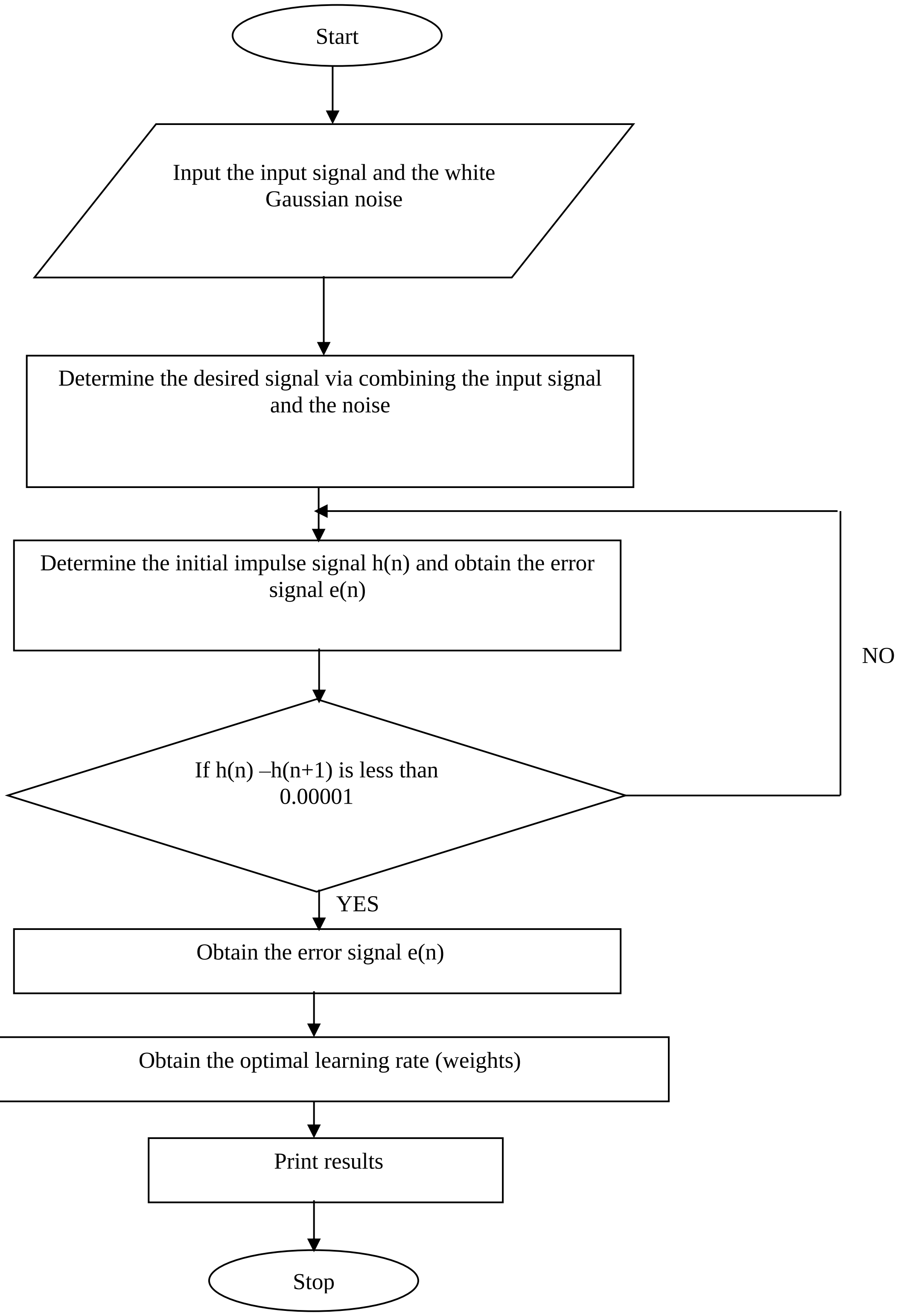

Figure 3: Flowchart for the NLMS algorithm

C. Sign Least Mean Square Model

The sign least mean square model used in this study is given as [32, 33, 34];

$$y(n) = w(n-1)u(n) \quad (14)$$

where: y(n) is the filtered output at step n, w(n) is the vector of filter weight estimates at step n, n is the current time index and u(n) is the vector of the buffered input samples at step n. The error e(n) is given as:

$$e(n) = d(n) - y(n) \quad (15)$$

where: e(n) is the estimated error at step n, d(n) is the desired response at step n and y(n) is the filtered output at step n. The weights of the filter w(n) was adjusted with the model:

$$w(n) = w(n-1) + f(u(n), e(n), \mu) \quad (16)$$

Where:

$$f(u(n), e(n), \mu) = \mu e(n) u^{*}(n) \quad (17)$$

μ is the adaptation step size, and $u^{*}(n)$ is the complex conjugate of the vector of buffered input samples at step n.

The sign least mean square (SLMS) model used are of three categories namely: Sign-Data Least Mean Square Model (SDLMS), sign-sign Least Mean Square (SSLMS) and Sign-Error Least Mean Square (SELSM) .

In the Sign-Data Least Mean Square (SD LMS) Model [35, 36], the filter weights are obtained using the LMS models, but each time the weights were updated, it replaces each sample of the input vector u(n) in equation 17 with +1 when the sample is positive, -1 when the sample is negative and zero when it is zero.

In the sign-sign Least Mean Square (SSLMS) Model [37, 38], the filter weights were obtained using the LMS models, but each time the weights were updated, it replaces each sample of the input vector u(n) in equation 17 with +1 when the sample is positive, -1 when the sample is negative and zero when it is zero. It also replaces e(n) with +1 when the sample is positive, -1 when the sample is negative and zero when it is zero.

In the Sign-Error Least Mean Square (SELSM) Model [39] the filter weights are obtained using the LMS models, but each time the weights were updated, it replaces each sample of the input vector e(n) in equation 17 with +1 when the sample is positive, -1 when the sample is negative and zero when it is zero.

III. METHODOLOGY

A. Proposed Modified Least Mean Square Model

The proposed least mean square model was done by combining the iteration conditions and stopping criterions of both least mean square model and normalized least mean square model shown from equation 18 to equation 22. The flowchart for the proposed modified least mean square adaptive filter is shown in Figure 4.

International Multilingual Journal of Science and Technology (IMJST)
ISSN: 2528-9810
Vol. 3 Issue 12, December - 2018

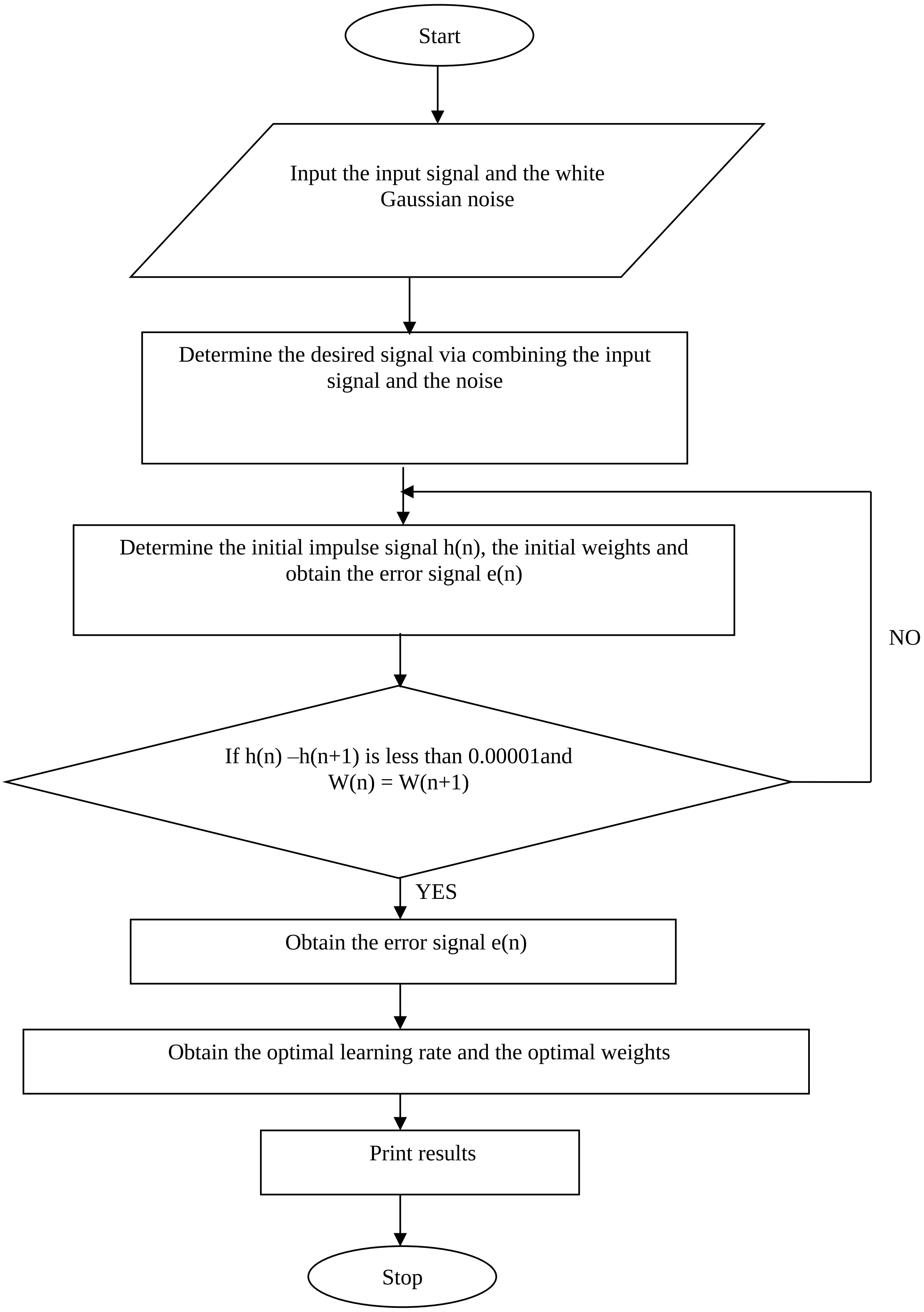

Figure 4: Flowchart for the proposed LMS model

$$\Delta W(n) = -\mu\left(\frac{\delta E\left(e^2(n)\right)}{dW(n)}\right) \quad (18)$$

Thus:

$$W(n+1) = W(n) - \mu\left(\frac{\delta E\left(e^2(n)\right)}{dW(n)}\right) \quad (19)$$

$$X(n) = [x(n), x(n-1), \dots, X(n-L+1)]^T \quad (20)$$

$$e(n) = d(n) - h^H(n)x(n) \quad (21)$$

$$h(n+1) = h(n) + \left(\frac{\mu e^*(n)x(n)}{X^H(n)x(n)}\right) \quad (22)$$

where: $\Delta W(n)$ is an incrementing factor, n is the time, W(n) is the total weights to be adjusted at time n, X(n) is the total input signal at time n, w(n) and x(n) are the weights and input signals of each adaptive filter coefficients respectively, L is the filter order, e(n) is the error of the current sample n, d(n) is the desired signal of the current sample n, h(n) is the impulse response and μ is the learning rate of the system.

B. Performance Measures for the Adaptive Filter Models

Root mean Square Error (RMSE) and Sum of Square Error (SSE) of the desired signal and the output signal were determined with the models in equation (23) and equation (24) respectively.

$$RMSE = \sqrt{\frac{1}{n}\sum(desired - output)^2} \quad (23)$$

$$SSE = \sum(desired - output)^2 \quad (24)$$

Where n is the number of iterations.

Also, R-square values was computed. The parameters were used to evaluate the performance of the various adaptive filter model presented in this paper.

IV SIMULATION AND RESULTS

The input signal (or desired signal) in Figure 5 is from Hallelujah chorus by Handel whereas Gaussian noise was used as noise signal. The input signal from Hallelujah chorus by Handel was recorded and uploaded in Matlab 7.9 and mixed with the white Gaussian noise. The mixture of the input signal and noise signal was subjected to filtering using five least square algorithms namely; LMS, NLMS, SSLMS, SDLMS and SELMS models. Also the proposed modified LMS algorithm was used to cancel out noise from the desired signal (recorded Handel's Hallelujah chorus). Performance measures were used to ascertain the best adaptive filter model that cancelled out noise from the desired signal.

Up to 70114 sets of data were generated during the simulation of the song with the noise. The error between the input signal to the adaptive filter (speech signal with noise) and the output of the adaptive filter is obtained and plotted for each of the algorithms. In order to compare the performance and the speed of convergence of the algorithms the signal value versus the number of iterations graph is plotted and the following performance parameters are also computed for each algorithm; R-square values, sum of square error and root mean square error. The plot shows the number of iterations it takes for the mean square error to attain the recorded performance parameters values for each of the algorithms.

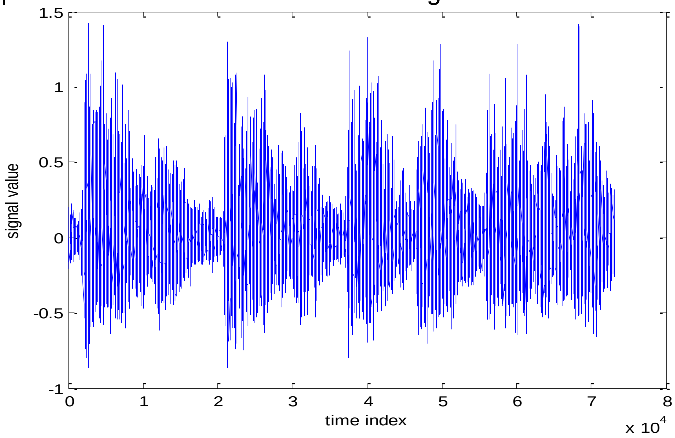

Figure 5: Signal Value versus Time Index or Iteration For The Desired Signal

After filtering with LMS model, the plot of the output signal value versus time index or iteration is given in figure 6 while the plot of the error signal value versus time index for the LMS model is given in figure 7. Figure 8 shows the plot of the combination of desired signal, output signal and error signal for the LMS model.

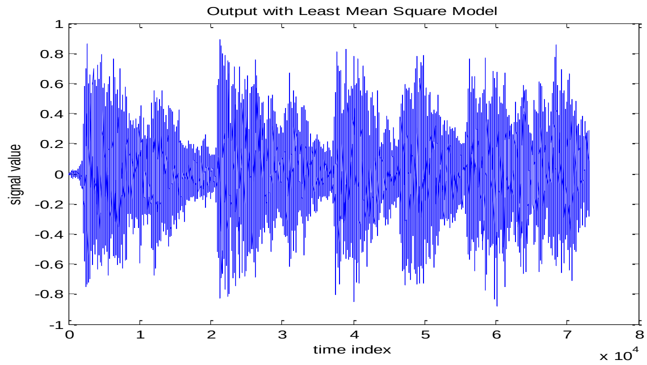

Figure 6: Output Signal Value versus Time Index or Iteration For The Output signal from LMS model

Figure 7 and figure 8 show that for the LMS model the error signal (the red line) tends toward zero after some iteration which affirms the postulates that when the desired signal is subtracted from the output signal, the error will be zero. Particularly, the more error tends to zero, the more the noise is cancelled by the LMS model. When the filtering process occurs, the coefficients of the adaptive filter were adjusted by the LMS model. The actual and the estimated coefficients of the LMS filter are shown in Figure 9. It can be seen from Figure 9 that there is no much deviation of the estimated coefficients from the actual coefficients (the actual coefficients are the coefficients of the adaptive filter originally present before tuning, while the estimated coefficients are the final coefficients obtained after simulation by the LMS model).

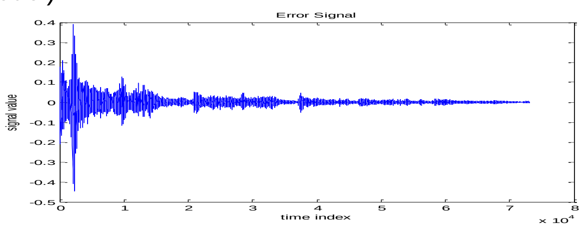

Figure 7: Error Signal Value versus Time Index or Iteration For The Output signal from LMS model

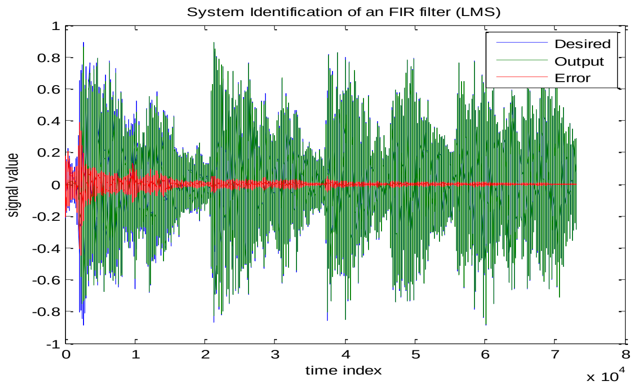

Figure 8: Plot showing the combination of desired signal, output signal and error signal for the LMS model

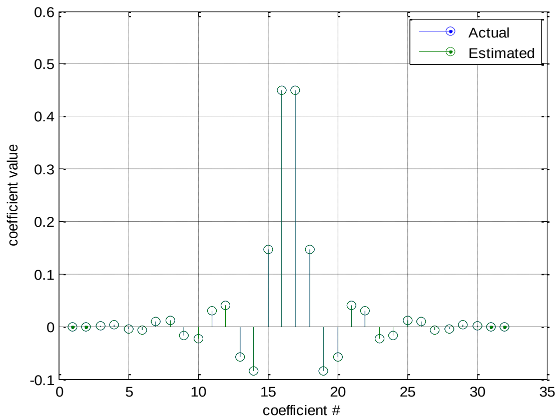

Figure 9: Plot showing the adaptive filter coefficient output for the LMS model

Similarly, for the Normalized Least Mean Square (NLMS) model, the plot of the output signal, the error signal , the combination of desired signal, output signal and error signal and the coefficient output are given in figure 10 , figure 11 , figure 12 and figure 13 respectively.

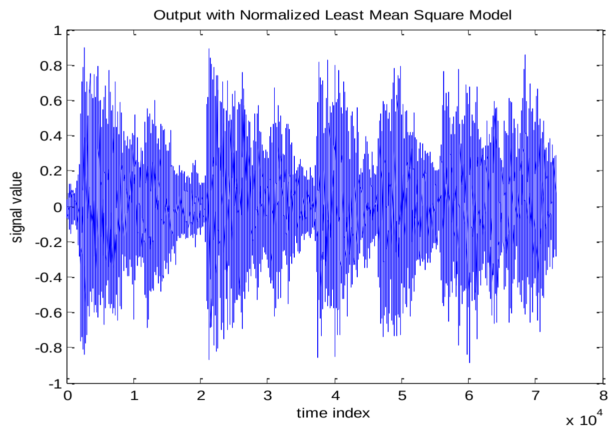

Figure 10: Output Signal Value versus Time Index or Iteration For The Output signal from NLMS model

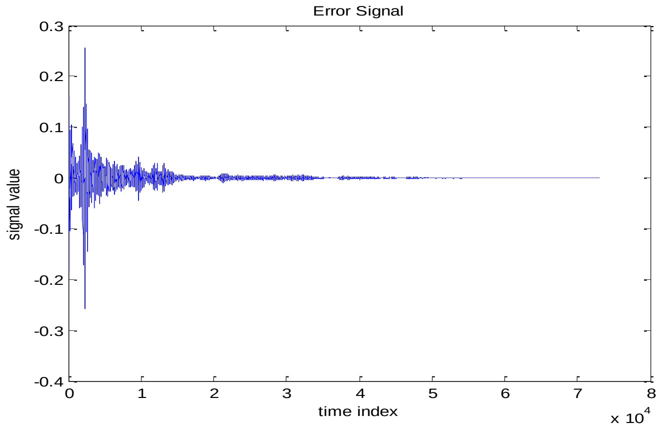

Figure 11: Error Signal Value versus Time Index or Iteration For The Output signal from NLMS model

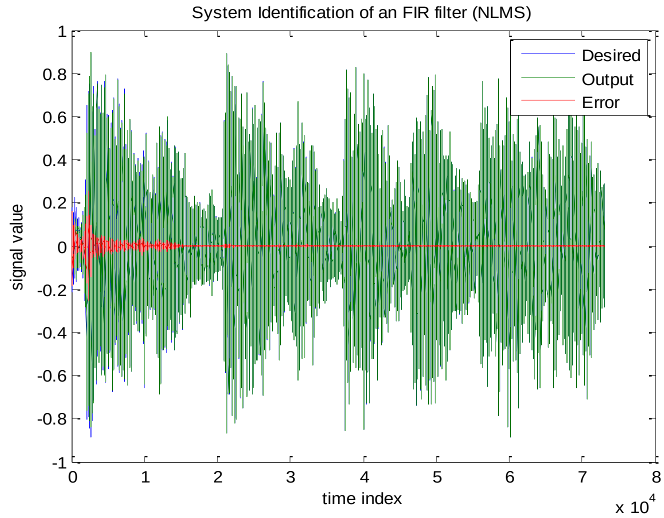

Figure 12: Plot showing the combination of desired signal, output signal and error signal for the NLMS model

Although the results of the LSM and that of the NLSM look alike, it will be observed from the error signal plots that the error plot of figure 11 for the NLSM algorithm converged (tends towards zero) faster than that (figure 7) of the LMS algorithm.

Again, similar graph plots are obtained for the other three models, namely; SSLMS, SDLMS and SELMS models. However, in view of the close similarities among their graph plots only the plots for the Sign-Sign Least Mean Square (SSLMS) model are given. Accordingly, for the SSLMS model, the plot of the output signal, the error signal , the combination of desired signal, output signal and error signal and the coefficient output are given in figure 14 , figure 15 , figure 16 and figure 17 respectively.

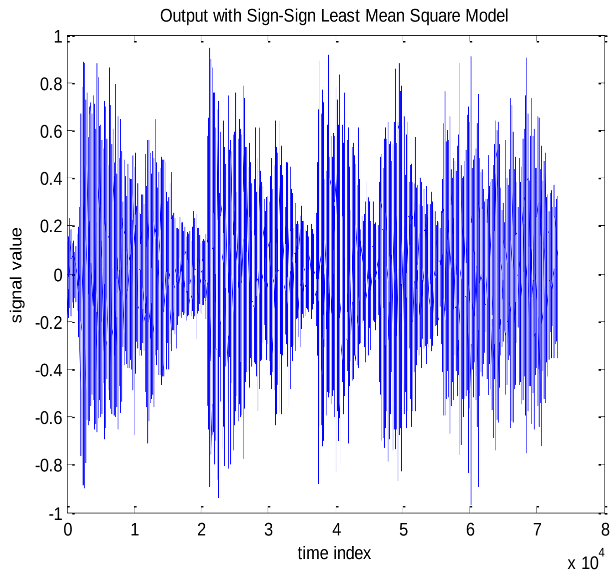

Figure 14: Output Signal Value versus Time Index or Iteration For The Output signal from SSLMS model

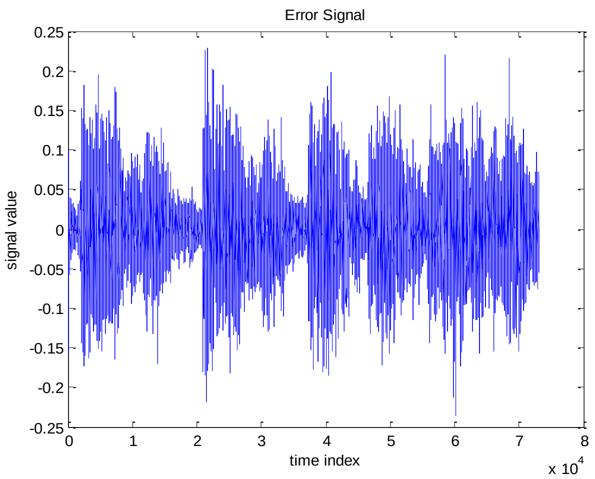

Figure 15: Error Signal Value versus Time Index or Iteration For The Output signal from SSLMS model

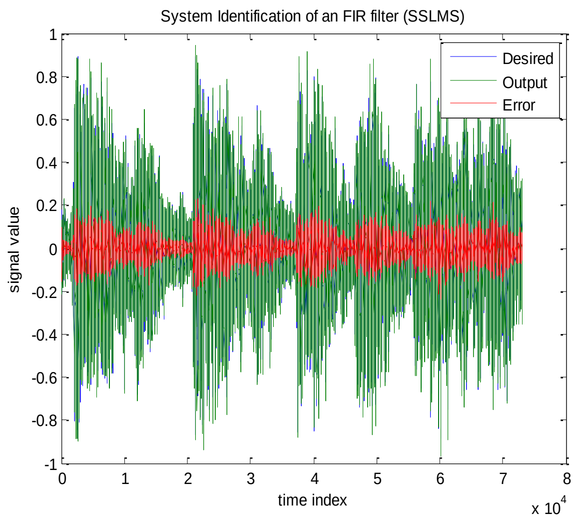

Figure 16: Plot showing the combination of desired signal, output signal and error signal for the SSLMS model

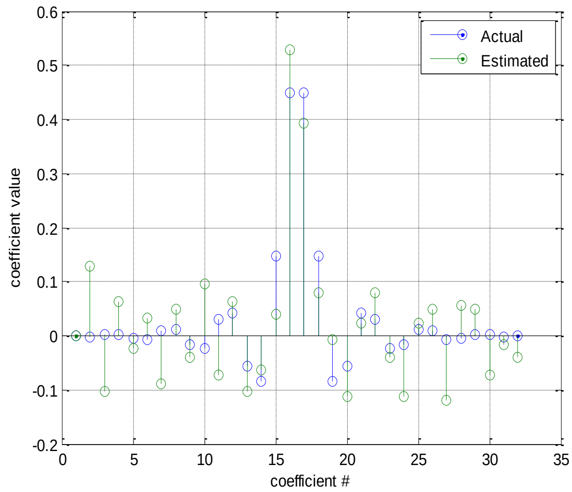

Figure 17: Plot showing the adaptive filter coefficient output for the SSLMS model

The error signal plot (figure 16) shows that the error signal in the SSLMS model is much when compared with those of the LMS and NLMS models. Also, in the SSLMS model plot of figure 17 there is a clear difference between the estimated coefficients from the actual coefficients unlike those of the LMS and NLMS models that is no marked difference between the estimated coefficients and the actual coefficients.

Furthermore, for the proposed modified LMS Model, the plot of the output signal, the error signal , the combination of desired signal, output signal and error signal and the coefficient output are given in figure 18 , figure 19 , figure 20 and figure 21 respectively.

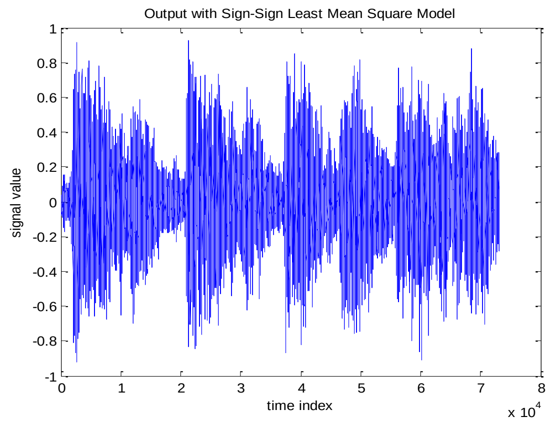

Figure 18: Output Signal Value versus Time Index or Iteration For The Output signal from the proposed modified LMS model

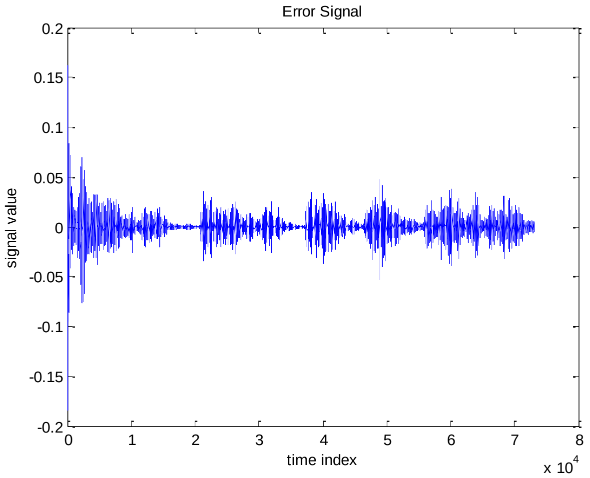

Figure 19: Error Signal Value versus Time Index or Iteration For The Output signal from the proposed modified LMS model

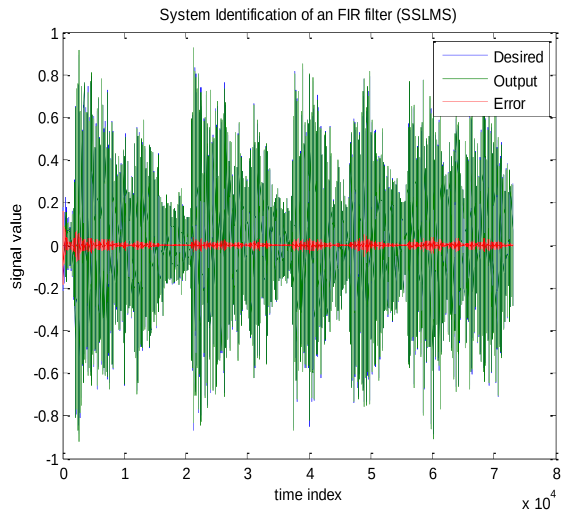

Figure 21: Plot showing the combination of desired signal, output signal and error signal for the proposed modified LMS model

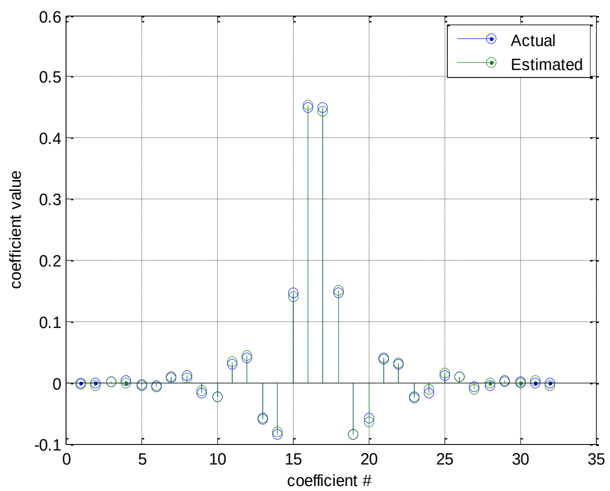

Figure 21: Plot showing the adaptive filter coefficient output for the proposed modified LMS model

Generally, the results for the proposed modified LMS model are similar to that of the LSM and NLSM model, however, the subtle differences in their performance and obvious from the R-square values, sum of square error and root mean square error of the different models , as shown in Table 1.

Table 1: Summary of R-square values, sum of square error and root mean square error of the different models

S/N	MODELS	R-SQUARE VALUES	SUM OF SQUARE ERROR	ROOT MEAN SQUARE ERROR
1	LMS	0.9889	29.9311	0.2020
2	NLMS	0.9981	5.0112	0.0083
3	SDLMS	0.9888	30.1627	0.2053
4	SELMS	0.9978	5.8741	0.0202
5	SSLMS	0.9630	99.9945	0.0370
6	Proposed Modified LMS	0.9983	5.0011	0.0081

From the results in Table 1, the proposed modified LMS model is the best model for cancelling noise since it has the highest R-square value of 0.9983, the lowest SSE value of 5.0011 and the lowest RMSE value of 0.0081. The next model in performance rating is the NLMS with R-square value of 09981, SSE value of 5.0112 and RMSE value of 0.0083. On the other hand, the sign-sign least mean square (SSLMS) model has the highest sum of square error value of 99.9945. This implies that it is inefficient to use SSLMS to cancel noise in the given case considered in this paper.

Finally, the model with the highest R-square value was the proposed LMS model, followed by normalized least mean square. This makes it the most efficient models for cancelling out noise in the Hallelujah chorus that was recorded. Since both the proposed modified LMS model and the NLMS model have RMSE of the acceptable range of 0 to10%, both can be used to eliminate noise in the song.

V. CONCLUSION

Five adaptive filter models namely; Least mean square (LMS), normalized least mean square (NLMS), Sign-data least mean square (SDLMS), sign-error least mean square (SELMS) and sign-sign least mean square (SSLMS) models were studied and a modified LMS algorithm was developed. The six adaptive filter models were then used to cancel out the white Gaussian interference in the recording of Handel’s Hallelujah chorus. The simulations were conducted using Mathlab. The results showed that the sign-sign least mean square model had the highest sum of square error and root mean square error and the lowest R-Square value which made it the least preferred model in the adaptive filtering of the noise in Handel’s Hallelujah chorus. The newly developed modified LMS adaptive filter algorithm is the best algorithm among the six adaptive filter algorithms studied; it had the lowest sum of square error and root mean square error and the highest R-Square value. In any case, both the NLMS and the newly developed modified LMS adaptive filter algorithm have acceptable root mean square error values; hence both algorithms can be used to eliminate noise in the Handel's Hallelujah chorus.

REFERENCE

1. Kaurav J., Narwaria R.P., (2017) A Review on Application of Adaptive Algorithms in Signal Processing International Journal of Innovative Research in Computer and Communication Engineering. Vol. 5, Issue 6, June 2017
2. Dhiman, J., Ahmad, S., & Gulia, K. (2013). Comparison between Adaptive filter Algorithms (LMS, NLMS and RLS). *International Journal of Science, Engineering and Technology Research (IJSETR)*, *2*(5), 1100-1103.
3. Zhao, S. (2009). Performance analysis and enhancements of adaptive algorithms and their applications. *PhD, School of computer engineering, Nanyang technological university*.
4. Gurrapu, O. (2009). *Adaptive filter algorithms for channel equalization*(Master's thesis, Magisteruppsats).
5. Shynk, J. J. (1992). Frequency-domain and multirate adaptive filtering. *IEEE Signal Processing Magazine*, *9*(1), 14-37.
6. Gupta, P., & Beniwal, P. (2015). Adaptive filters algorithms: a performance comparison. *Int. J. Eng. Res. Gen. Sci*, *3*(4), 2091-2730.
7. Telagarapu, P., Biswal, B., & Prasad, P. M. K. (2012, February). Fast convergence and error free SAF in Multimedia Applications. In *Computing, Communication and Applications (ICCCA), 2012 International Conference on*(pp. 1-5). IEEE.
8. Apolinário Jr, J. A., & Netto, S. L. (2009). Introduction to adaptive filters. *QRD-RLS Adaptive Filtering*, *1978*, 23.
9. Petraglia, M. R., & Torres, J. C. (2002). Performance analysis of adaptive filter structure employing wavelet and sparse subfilters. *IEE Proceedings-Vision, Image and Signal Processing*, *149*(2), 115-119.
10. Douglas, S. C. (1999). Introduction to adaptive filters. *Digital signal processing handbook*, 7-12.
11. Widrow, B., & Winter, R. (1988). Neural nets for adaptive filtering and adaptive pattern recognition. *Computer*, *21*(3), 25-39.
12. Prasad, S. R., & Godbole, B. B. (2017). Optimization of LMS Algorithm for System Identification. *arXiv preprint arXiv:1706.00897*.
13. Vijay, J. P., & Sharma, N. K. (2014). Performance analysis of RLS over LMS algorithm for MSE in adaptive filters. *International Journal of Technology Enhancements and Emerging Engineering Research*, *2*(4), 40-44.
14. Diniz, P. S. (2013). The least-mean-square (LMS) algorithm. In *Adaptive Filtering* (pp. 79-135). Springer US.
15. Farhang-Boroujeny, B. (2013). LMS algorithm. *Adaptive Filters: Theory and Applications*, 139-206.
16. AbdulKabir, A. A., Aibinu, A. M., Onwuka, E. N., & Salami, M. J. E. (2013). New Method of LMS Variable Step-Size Formulation for Adaptive Noise Cancellation.
17. Lau, Y. S., Hussian, Z. M., & Harris, R. (2003, December). Performance of adaptive filtering algorithms: A comparative study. In *Australian Telecommunications Networks and Applications Conference*.
18. Sharma, R. A. G. H. A. V. E. N. D. R. A., & Pyara, V. P. (2012). A comparative analysis of mean square error adaptive filter algorithms for generation of modified scaling and wavelet function. *Int J Eng Sci Technol*, *4*(4), 1396-1401.
19. Walach, E., & Widrow, B. (1984). The least mean fourth (LMF) adaptive algorithm and its family. *IEEE transactions on Information Theory*, *30*(2), 275-283.
20. Turan, C., & Amirgaliev, Y. (2016, September). A Robust Leaky-LMS Algorithm for Sparse System Identification. In *International Conference on Discrete Optimization and Operations Research* (pp. 538-546). Springer International Publishing.
21. Torti, F., Perrotta, D., Atkinson, A. C., & Riani, M. (2012). Benchmark testing of algorithms for very robust regression: FS, LMS and LTS. *Computational statistics & data analysis*, *56*(8), 2501-2512.
22. Bickel, D. R., & Fruehwirth, R. (2005). On a fast, robust estimator of the mode: Comparisons to other robust estimators with applications. *arXiv preprint math/0505419*.
23. Gu, Y., Tang, K., & Cui, H. (2004). LMS algorithm with gradient descent filter length. *IEEE Signal Processing Letters*, *11*(3), 305-307.
24. *Haykin S.* (2002) *Adaptive Filter Theory*, *Prentice Hall*, *2002*,. *ISBN 0-13-048434-2*.
25. Wang, L., & de Lamare, R. C. (2013). Set-Membership Constrained Conjugate Gradient Beamforming Algorithms. *arXiv preprint arXiv:1302.0422*.
26. Rai, A., & Kumar, A. (2015). *Analysis and design of adaptive volterra filters for system identification in time-varying environment* (Doctoral dissertation).
27. Singh, A. (2001). Adaptive noise cancellation. *Central Elektronica Engineering Research Institute, University of Dehli*.
28. Yuan, K., Ying, B., & Sayed, A. H. (2016). On the influence of momentum acceleration on online learning. *Journal of Machine Learning Research*, *17*(192), 1-66.
29. Feng, Y., Chen, F., Zeng, R., Wu, J., & Shu, H. (2015). Error Gradient-based Variable-Lp

Norm Constraint LMS Algorithm for Sparse System Identification. *arXiv preprint arXiv:1509.07951*.
30. Arablouei, R., Werner, S., & Dogancay, K. (2014). Analysis of the Gradient-Descent Total Least-Squares Adaptive Filtering Algorithm. *IEEE Trans. Signal Processing*, *62*(5), 1256-1264.
31. Chandra, G. V. P., Yadav, S., & Krishna, B. A. (2014). Study of different adaptive filter algorithms for noise cancellation in real-Time environment. *International Journal of Computer Applications*, *96*(10).
32. Eng, K., Chen, Y. Y., & Kiang, J. E. (2009). *User's Guide to the Weighted-Multiple-Linear Regression Program (WREG Version 1.0)*. US Department, of the Interior, US Geological Survey.
33. Miller, S. J. (2006). The method of least squares. *Mathematics Department Brown University*, 1-7.
34. Hubscher, P. L., & Bermudez, J. C. M. (2002). An improved stochastic model for the least mean fourth (LMF) adaptive algorithm. In *Circuits and Systems, 2002. ISCAS 2002. IEEE International Symposium on* (Vol. 1, pp. I-I). IEEE.
35. Paul, B., & Mythili, P. (2012, August). ECG noise removal using GA tuned sign-data least mean square algorithm. In *Advanced Communication Control and Computing Technologies (ICACCCT), 2012 IEEE International Conference on* (pp. 100-103). IEEE.
36. Mythili, P., & Baby, P. (2012). ECG Noise Removal using GA tuned Sign-Data Least Mean Square Algorithm.
37. Zhang, T., & Gui, G. (2016). Sign Function Based Sparse Adaptive Filtering Algorithms for Robust Channel Estimation under Non-Gaussian Noise Environments. *Algorithms*, *9*(3), 54.

Armbrust, M., Fox, A., Griffith, R., Joseph, A. D., Katz, R. H., Konwinski, A., ... & Zaharia, M. (2009). *Above the clouds: A berkeley view of cloud computing* (Vol. 17). Technical Report UCB/EECS-2009-28, EECS Department, University of