International Journal of Telecommunications & Emerging Technologies
Vol. 1: Issue 2
www.journalspub.com

# Adaptive Noise Cancellation Using Normalized LMS Algorithm

*Saurabh R. Prasad*[1*], *Neeraj R. Prasad*[2], *Rohini P. Onkare*[3], *Yashwant V. Joshi*[4]
[1]Department of Electronics and Telecommunication Engineering, DKTE Society's Textile and Engineering Institute, Ichalkaranji, India
[2]Department of Nanoscience and Technology, Shivaji University, Kolhapur, India
[3]Department of Electronics and Telecommunication Engineering, Padmavibhushan Vasantdada Patil Institute of Technology, Budhgaon, Sangli, India
[4]Department of Electronics and Telecommunication Engineering, SGGS Institute of Engineering and Technology, Nanded, Maharastra, India

***Abstract***

*An adaptive filter is a digital filter that self-adjusts its transfer function according to an optimizing algorithm. There are a number of optimizing algorithms available, but the performance, simplicity, and stability of least mean square (LMS) algorithms outweigh other algorithms, thus LMS algorithm has become increasingly popular. The digital filter structure used in adaptive filtering is usually FIR with transversal or tapped delay line realization. Adaptive filtering is one of the core technologies in digital signal processing and finds diversified applications such as echo cancellation, channel equalization, system identification, adaptive noise cancellation, and adaptive beamforming. This paper presents Simulink implementation of an adaptive filter using normalized least mean square (NLMS) algorithms to reduce unwanted noise and thus improving signal quality.*

**Keywords:** adaptive filtering, noise cancellation, normalized LMS algorithm, Simulink, transversal filter

***Corresponding Author**
*E-mail: saurabh_prasad21@rediffmail.com*

## INTRODUCTION

An adaptive filter possesses the property of self-adjusting its transfer function according to an optimizing algorithm. Thus, an adaptive filter adapts its response as the input signal characteristics changes. Due to this demanding feature and the construction flexibility, the adaptive filters have been employed in many different applications that include telephonic echo cancellation, radar signal processing, navigation systems, channel equalization, biomedical signals processing, and many more.[1,2] As the power of digital signal processors has increased accompanied with decrease in its price adaptive filters have become much more common and are now routinely used in devices such as mobile phones, modem, camcorders, digital cameras, and medical monitoring equipment.[3]

Adaptive algorithm allows the filter to learn the initial statistics of the input signal and to track them afterwards for any further changes.[4] Thus, adaptive filters self-learn and that is called as its training phase. As time passes, the adaptive filter coefficients adjust themselves to minimize error signal and achieve the desired result. In certain applications where signal and noise occupy the same frequency band, e.g., in analyzing fetus electrocardiogram, we cannot use fixed digital filters but only adaptive filters. Similarly in applications where the input signal statistics, or noise statistics are changing with time, designing fixed filter is worthless but adaptive filters are the only solution.[5] The adaptive filter is a closed loop system and employs feedback in the form of an error signal. This feedback is used to tune its transfer function in order to minimize cost function which is a criterion for optimum performance of the filter. The adaptive filtering algorithm determines how to modify filter transfer function to minimize the cost on the next iteration. The most commonly employed cost function is the mean square of the error signal (MSE).[6]

## ADAPTIVE FILTER FOR NOISE CANCELLATION

Noise is a nuisance or disturbance during communication and it is unwanted. Noise occurs because of many factors and few important factors of these are interference, delay, and overlapping.[7] The noise cancellation with the help of adaptive filter can be employed for variety of practical applications like denoising electrocardiograph, speech signals enhancement, and cancelling of side-lobes interference of an antenna array. The problem of controlling the noise level has become the focus of a tremendous amount of research over the years. In the process of transmission of information from the source to receiver side, noise from the surroundings automatically gets added to the signal. Similarly, the acoustic noise picked up by microphone is undesirable, as it reduces the perceived quality or intelligibility of the audio signal. The problem of effective removal or reduction of noise is a promising area of research. The use of adaptive filters is one of the most popular solutions to reduce the signal corruption caused by predictable and unpredictable noise.

### Active Noise Cancellation

One of the interesting applications of adaptive filters is called active noise cancellation (ANC). This is a technique to reduce the unwanted acoustic noise by generating anti-noise sound through a noise-cancelling speaker. Anti-noise is a sound wave with the same amplitude, but with opposite phase compared to the original noise. The unwanted noise and anti-noise superimpose acoustically those results in noise cancellation.[8] The idea of implementation of ANC is shown in Figure 1. Although the concept ANC has been around for more than 75 years, it is still an active and important research topic because of its wide range of applications that include:

(i) Provided that quiet environment in automobile and aero plane cabins
(ii) Lessening noise from motors, heavy machinery, and engines
(iii) High-end headphone systems
(iv) Decreasing mechanical wear out and fuel consumption through vibration control
(v) Reducing background noise in communication systems, e.g., radio

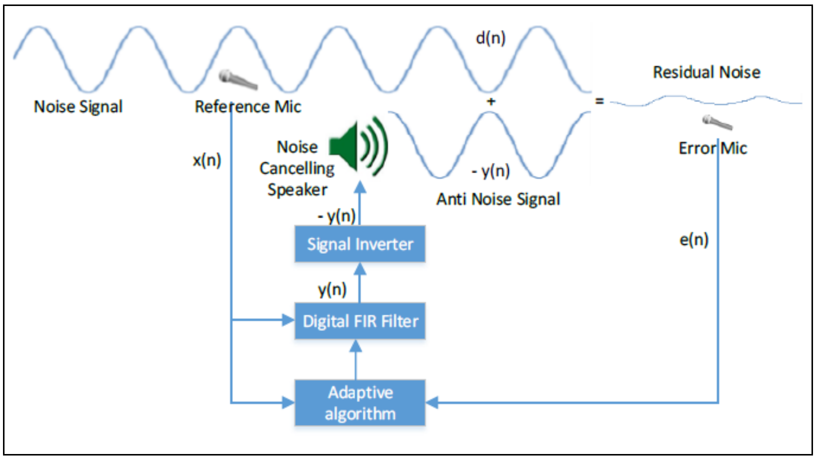


***Fig. 1.*** *Active Noise Cancellation Scheme.*

FIR filter in transversal structure is usually used as the filtering element in adaptive least mean square (LMS) filter. This structure is also called as tapped delay line and shown as in Figure 2. This figure indicates that the implementation of FIR filter requires delay line, adders, and multipliers only. Here, $w_0$, $w_1$,.. are filter coefficients. The order of the filter is L-l where L is the number of filter coefficients.

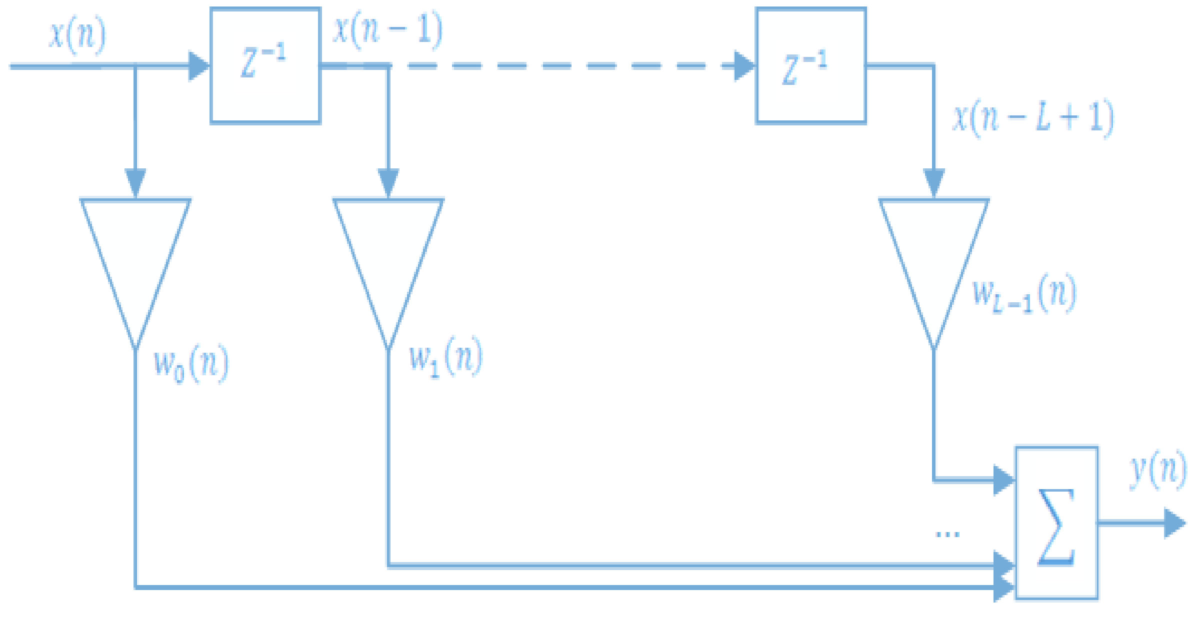


***Fig. 2.*** *FIR Filter as Tapped Delay Line (Transversal Structure).*

**Different Adaptive Filtering Algorithms**

The basic adaptive algorithms those are widely used include LMS, and the recursive least square (RLS). Among all adaptive algorithms LMS has probably become the most popular for its robustness, good tracking capabilities, and simplicity in stationary environment.[9] Although RLS is best suited for high convergence speed in non-stationary environment, but its computational complexity renders it less popular choice. Therefore, a tradeoff is required in convergence speed and computational complexity.

However, in such situations, LMS provides the right solution. A number of variants of LMS algorithm exist and they are described as standard LMS, leaky LMS, normalized LMS, sign LMS, sign-error, sign-data LMS algorithm, sign–sign LMS algorithm, fast block LMS, and many more. Each of these types has their own merits and demerits. However, the most popular of them are standard LMS and normalized LMS and they keep the major focus in this article.

Figure 3 shows a typical adaptive noise cancellation system. The adaptive filter processes the reference signal x(n) to produce the output signal y(n) by convolution with filter's weights w(n). Unlike fixed FIR filter, in adaptive filter the filter's weight changes continuously. The filter output y(n) is subtracted from desired signal d(n) to obtain an estimation error e(n). The objective here is to minimize the error signal e(n). This error signal is used to incrementally adjust the filter's weights for the next time instant.

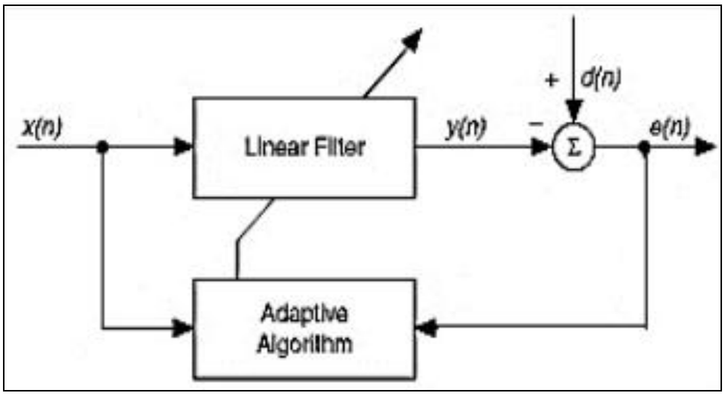


***Fig. 3.*** *General Schematic of Noise Cancellation Using Adaptive Filter*.

**LMS Algorithm**

According to estimation and detection theory, the optimum solution for adaptive filter weights given by Weiner, also called as Weiner–Hopf equation, is $w_{\text{opt}} = \boldsymbol{R}^{-1}\boldsymbol{P}$; where R is the autocorrelation matrix of input signal x(n) and P is the cross-correlation matrix between input signal x(n) and desired signal d(n).[10] Thus, the computation of optimal solution involves matrix inversion, which is computationally quite intensive. So, the alternate solution to this problem is instead of calculating the optimal solution $w_{\text{opt}}$, by iteration the approximate solution can be obtained. The algorithm starts by assuming small initial weights, zero in most cases, and by finding the gradient of the MSE cost function, the weights are updated iteratively at each step.

That is, if the MSE gradient is positive, it implies the error is increasing positively, which indicates to reduce the weights. In the same way if the gradient is negative, it indicates to increase the weights. So, the basic weight update equation is given as; $W_{n+1} = W_n - \mu\nabla\varepsilon[n]$ where $\varepsilon$ represents the mean-square error. The negative sign indicates that, we need to change the weights in a direction opposite to that of the gradient slope. From this equation, the weight updates equation of LMS algorithm can be derived as $w(n+1) = w(n) + 2\mu e(n)x(n)$.[11]

Gabor was the first to put forth the idea of a nonlinear adaptive filter in 1954 using a Volterra series but that could not become much popular. The legendry LMS algorithm was invented in 1959 by Stanford University professor Bernard Widrow and his first doctoral research scholar, Ted Hoff through their studies of pattern recognition. It has become one of the most widely used algorithms in adaptive filtering. LMS algorithm uses a stochastic gradient descent method in that the filter is only adapted based on the error at the current time. Thus, the LMS algorithm utilizes the gradient vector of the filter tap weights to converge to the optimal wiener solution. The LMS algorithm iteratively solves the Wiener–Hopf equation and finds the filter coefficients for an adaptive filter.

The mean-square error, as a function of filter weights is a quadratic function which means it has only one extreme that minimizes the MSE, and that is the optimal weight.[12] Thus, the LMS approaches near to these optimal weights by ascending/descending down the mean-square error versus filter weight curve. The LMS algorithm is based on the steepest descendent method from numerical optimization where the cost function is the squared error signal, i.e., $\varepsilon(n) = e^2(n)$. This error signal is fed back to the adaptive filter and its coefficients are changed by the specified algorithm in order to minimize some function of the error signal that is known as the cost function. LMS algorithm incorporates an iterative practice that makes successive corrections to the weight vector in the direction of the negative of the gradient vector which sooner or later leads to the minimum mean square error.

LMS algorithm is well recognized and extensively used due to its computational easiness. It is this easiness that has made it the standard against which all other adaptive filtering algorithms are judged. The LMS algorithm does not require matrix inversion $R^{-1}$. It does not require the availability of the autocorrelation matrix of the filter input and the cross correlation between the filter input and its desired signal. The LMS algorithms require fewer computational resources and memory than the RLS algorithms. The implementation of the LMS algorithms also is less complicated than the RLS algorithms. These benefits have contributed in making LMS algorithm the first choice by signal processing community.

***Normalized LMS Algorithm***

The normalized LMS (NLMS) algorithm is a modified form of the standard LMS algorithm. The main drawback of the standard LMS algorithm is that it is sensitive to the scaling of input vector x(n), thus the LMS algorithm suffers from slow and data dependent convergence behavior.[13] This makes it very hard to choose a step size μ that guarantees stability of the algorithm. The NLMS algorithm, an equally simple, but more robust variant of the LMS algorithm, exhibits a better balance between simplicity and performance than the LMS algorithm. The NLMS algorithm solves this problem by normalizing the input signal vector with the power of the input signal.[14] Due to this property the NLMS has been largely used in real-time applications. The weight update equation of the NLMS algorithm is given as below.

$$\bar{w}(n+1)=\bar{w}(n)+\frac{\mu e(n)x(n)}{x^{H}(n)x(n)}$$

## EXPERIMENTAL SETUP

The foremost objective of the noise cancellation is to evaluate the noise signal y(n) and to subtract it from original input signal plus noise signal d(n) and hence to obtain the noise free signal e(n). This method uses a primary input signal that contains the desired signal, i.e., speech signal plus noise and a reference input containing noise. The reference input is adaptively filtered and subtracted from the primary input signal to obtain the estimated error signal. The desired signal corrupted by an additive noise is recovered by an adaptive noise canceller using LMS algorithm. Thus, the adaptive noise canceller improves the SNR.

Simulink developed by Mathworks is a graphical programming environment for modeling, simulating, and analyzing dynamic systems. Its primary interface is a graphical object level block diagramming tool and a customizable set of Simulink block libraries. Using Simulink library browser, the desired block can be searched and put in new model file. It offers direct interaction with the Matlab environment and can either drive Matlab or be scripted from it.

Simulink is widely used in multidomain applications including digital signal processing for simulation and model-based design. Simulink provides a graphical editor, customizable block libraries, and solvers for modeling and simulating dynamic systems. Thus, Simulink enables to incorporate Matlab algorithms into models, and export simulation results to Matlab for further analysis.

In this paper, we present an implementation of NLMS algorithms on Simulink platform with the intention to noise cancellation. To begin with Simulink, on Matlab prompt "Simulink" is typed to open the same**.** If prior knowledge of the tap-weight vector w(n) is available, then we have to use it as initial condition $w_0$. Otherwise, set $w_0 = 0$. The Simulink implementation is given in Figure 4 which uses different blocks from Simulink library browser. These blocks include signal from workspace, digital filter, downsample, LMS filter, constant, random source, sum, and signal to workspace. Setting parameters of these blocks in Simulink is a very important issue which demands matching of data types used in different blocks. The parameter setting of these blocks has been done as shown in Figures 5–15.

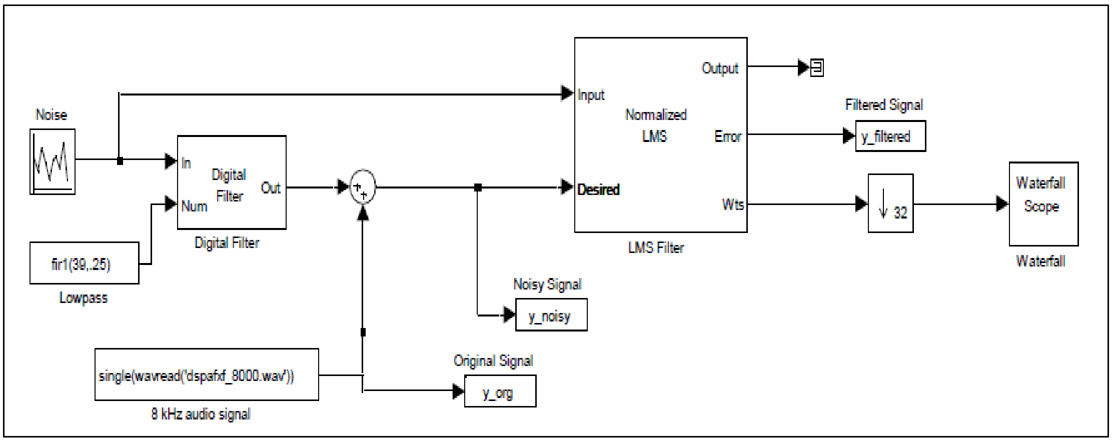


***Fig. 4.*** *Implementation of System Design for Adaptive Noise Using Simulink.*

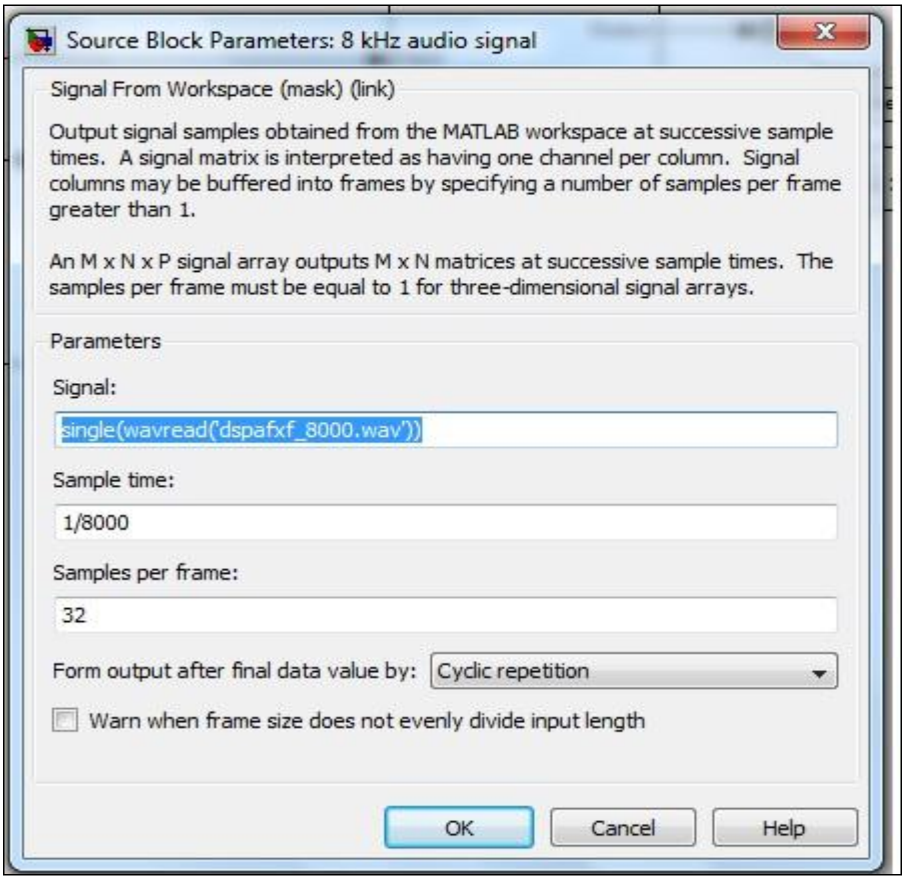


***Fig. 5.*** *Setting Parameters of 'Signal from Workspace Block'.*

 

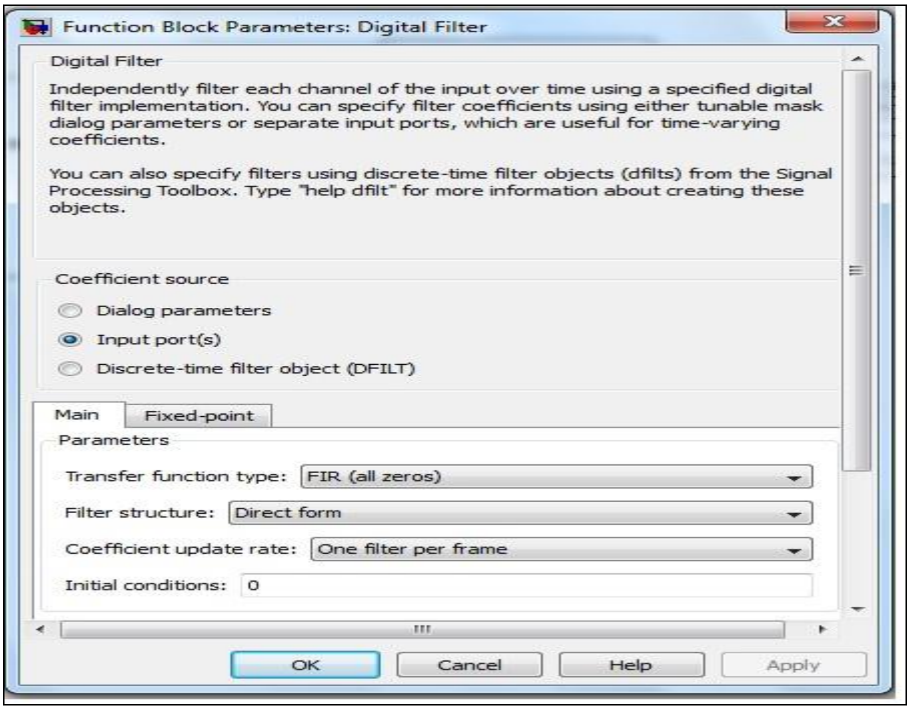


***Fig. 6.*** *Setting Main Parameters of 'Digital Filter Block'.*

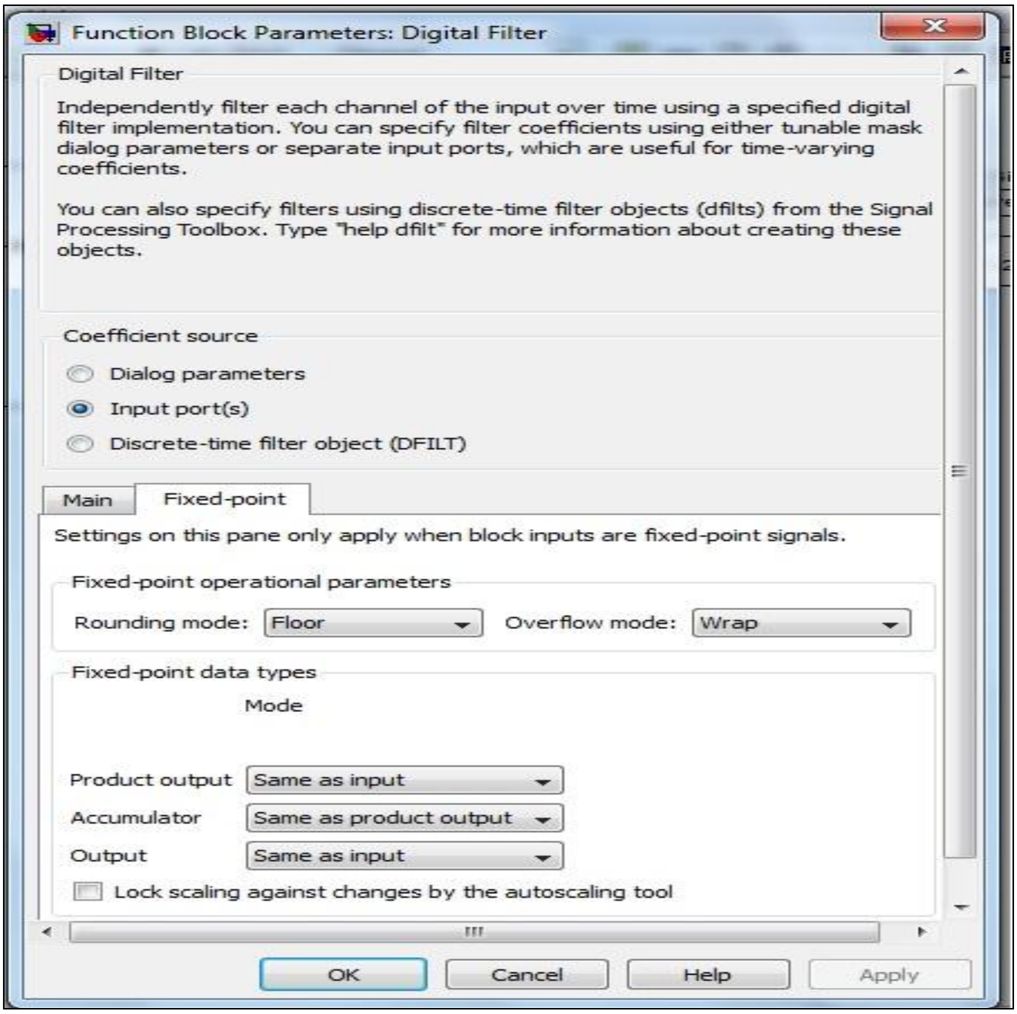


***Fig. 7.*** *Setting Fixed Point Parameters of 'Digital Filter Block'.*

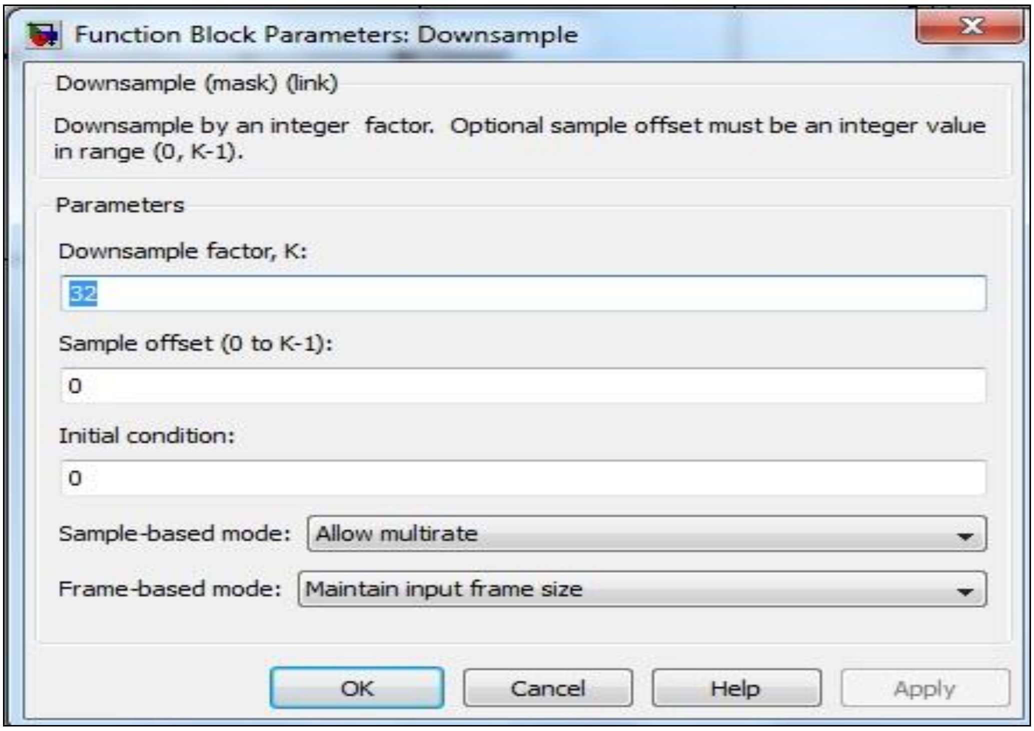


***Fig. 8.*** *Setting Parameters of 'Downsample Block'.*

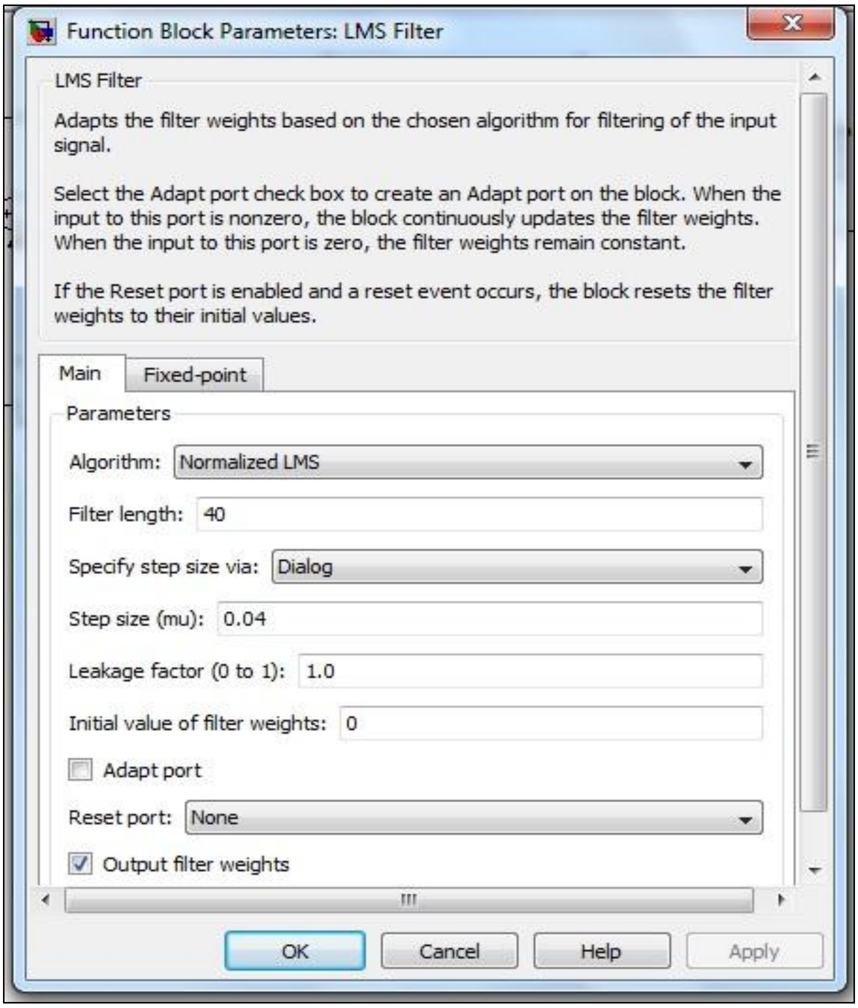


***Fig. 9.*** *Setting Main Parameters of 'LMS Filter Block'.*

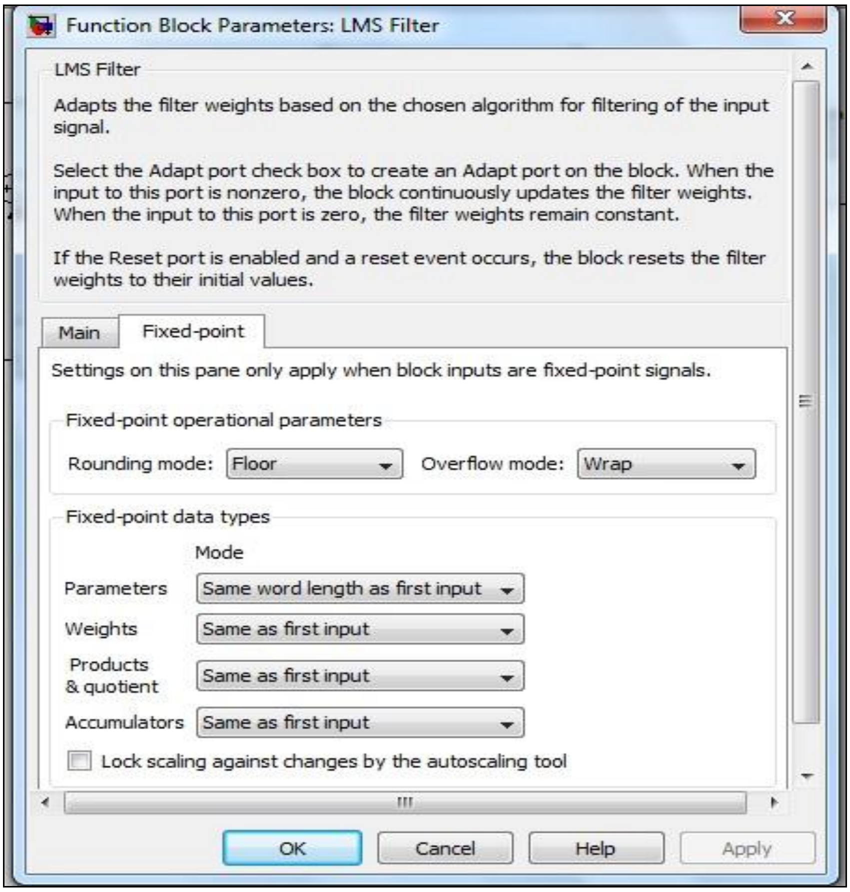


***Fig. 10.*** *Setting Fixed Point Parameters of 'LMS Filter Block'.*

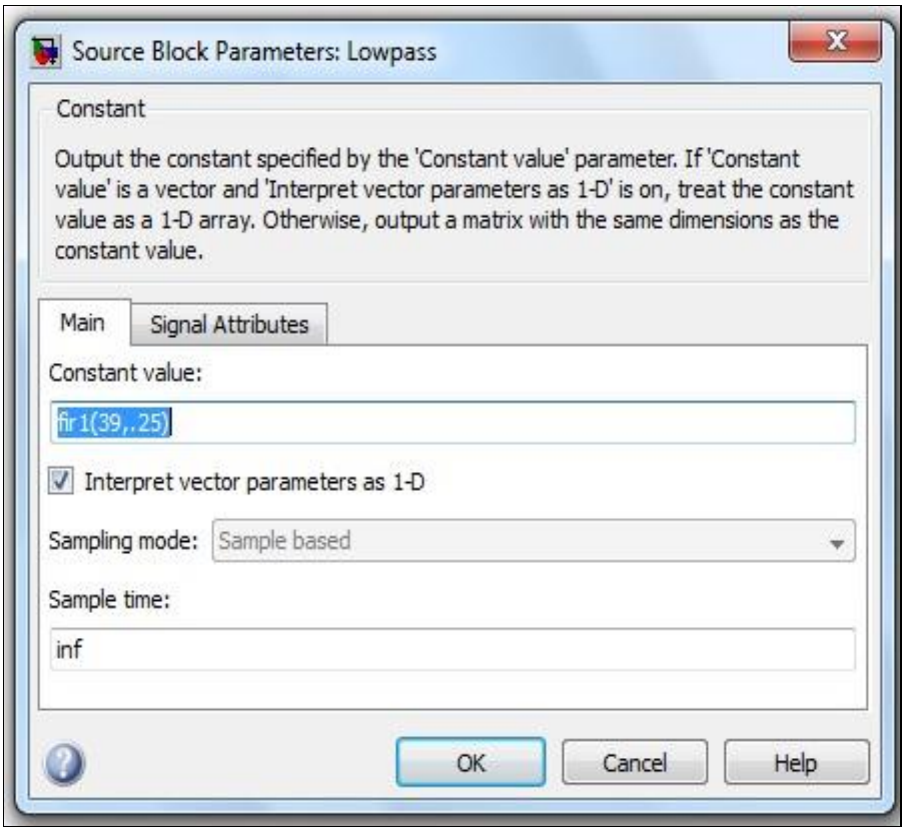


***Fig. 11.*** *Setting Main Parameters of 'Constant Block'.*

 

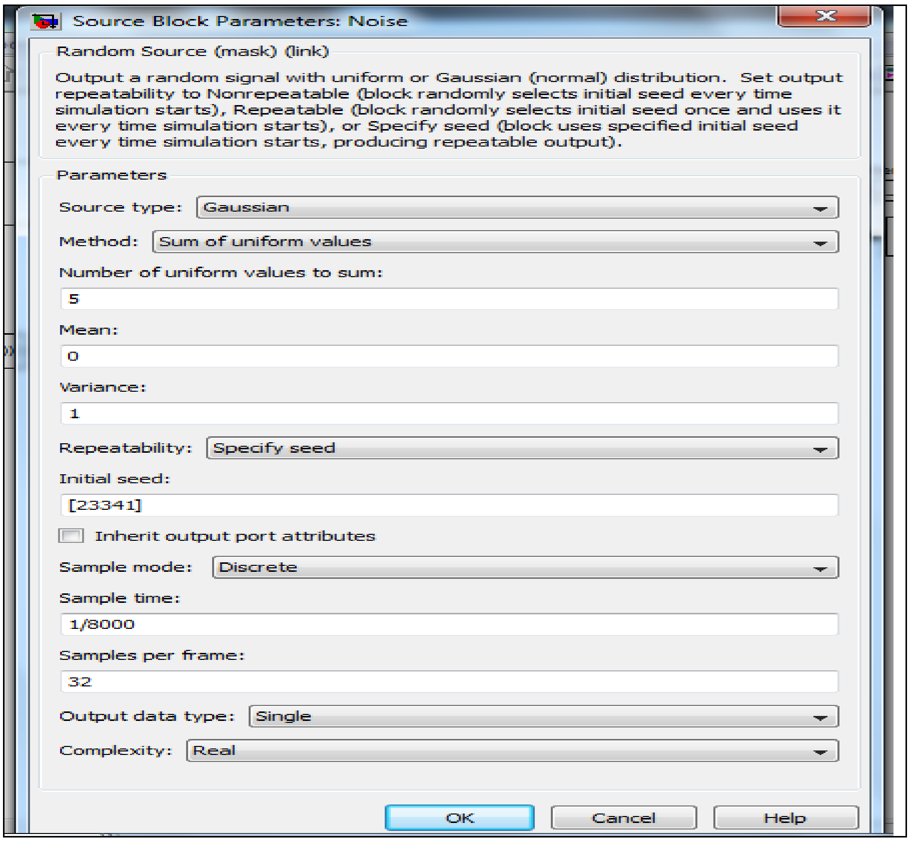


***Fig. 12.*** *Setting Main Parameters of 'Random Source Block'.*

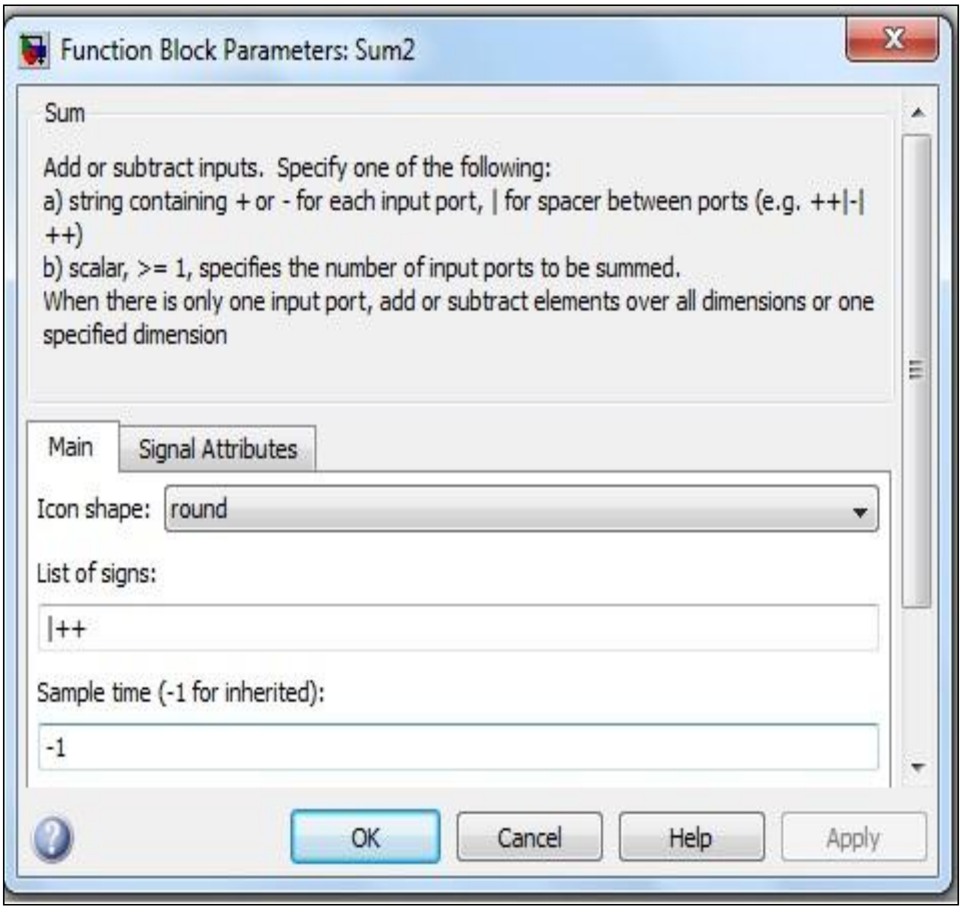


***Fig. 13.*** *Setting Main Parameters of 'Sum Block'.*

 

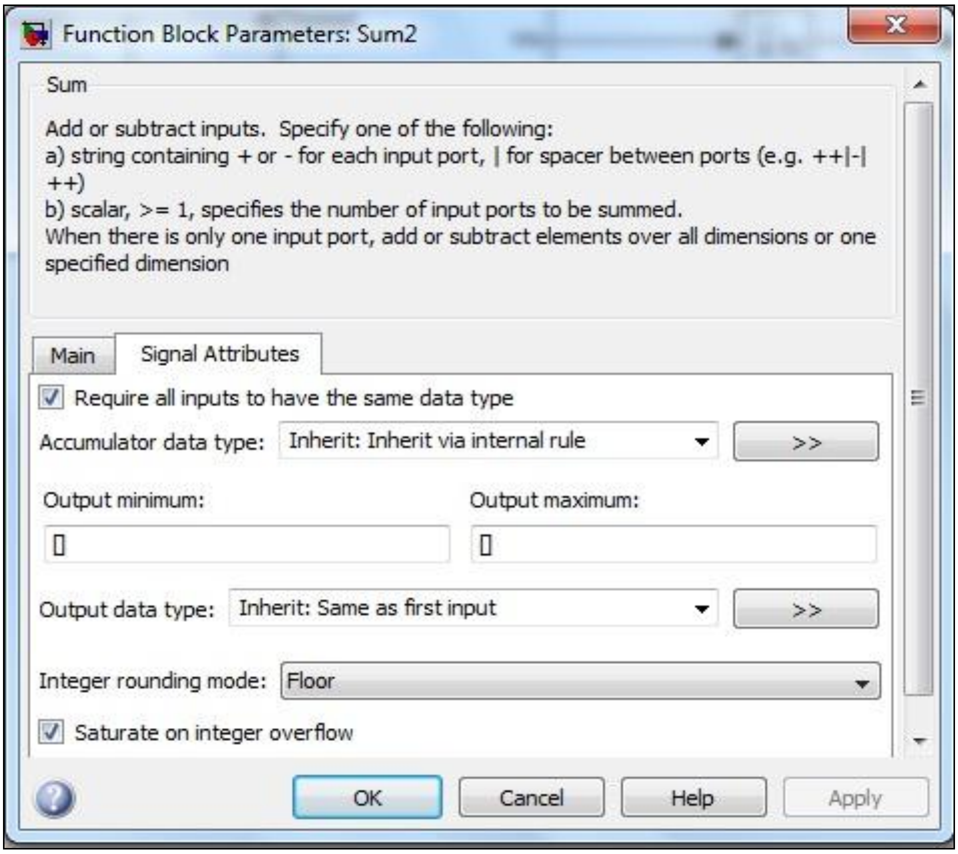


***Fig. 14.*** *Setting Signal Attributes Parameters of 'Sum Block'.*

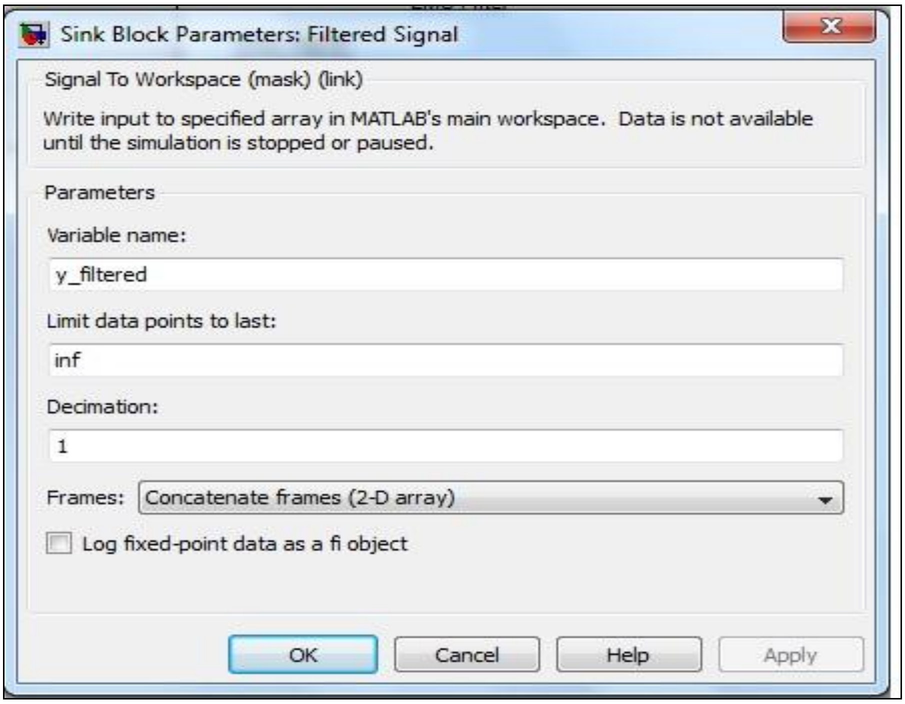


***Fig. 15.*** *Setting Parameters of 'Signal to Workspace Block'.*

## RESULT

After running the model, three different signals viz. original signal, noisy signal to be filtered which is also called as desired signal and filtered signal which is also called as error signal are available in the

Matlab workspace. They are plotted using Matlab command typed in command window as >>Figure (1), plot (y_oroginal), Figure (2), plot (y_noisy), Figure (3), and plot (y_filtered) and are plotted in Figures 16–18, respectively. The implementation result indicates that the desired signal is effectively filtered to recover the original sound and filter out the noise.

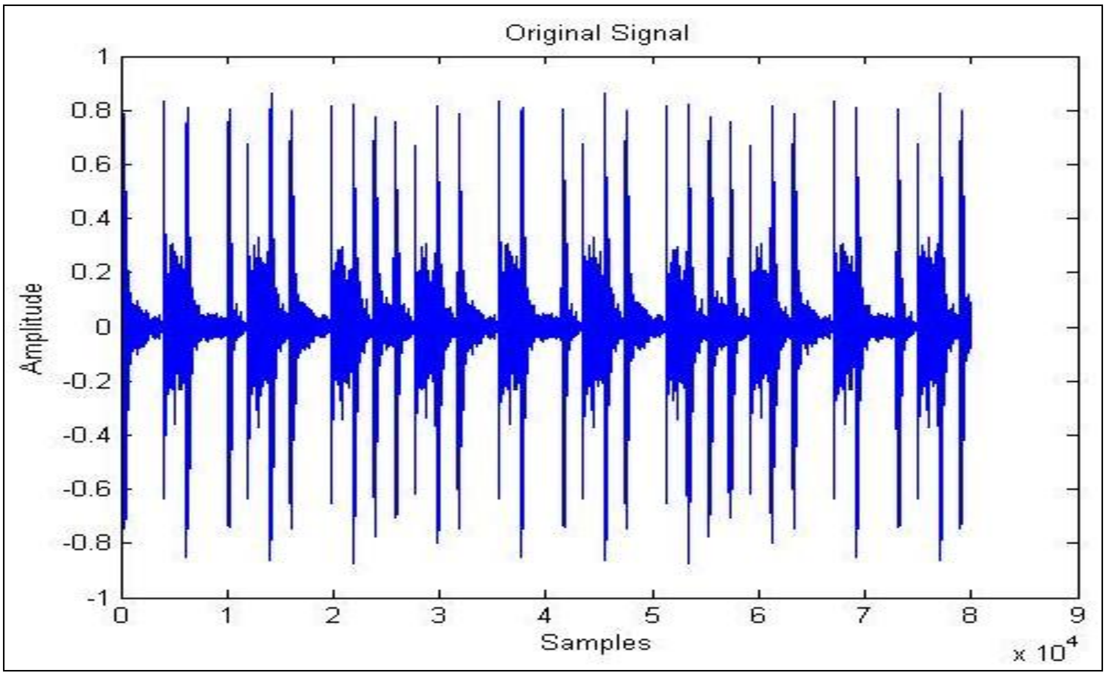

***Fig. 16***. *Original Signal*.

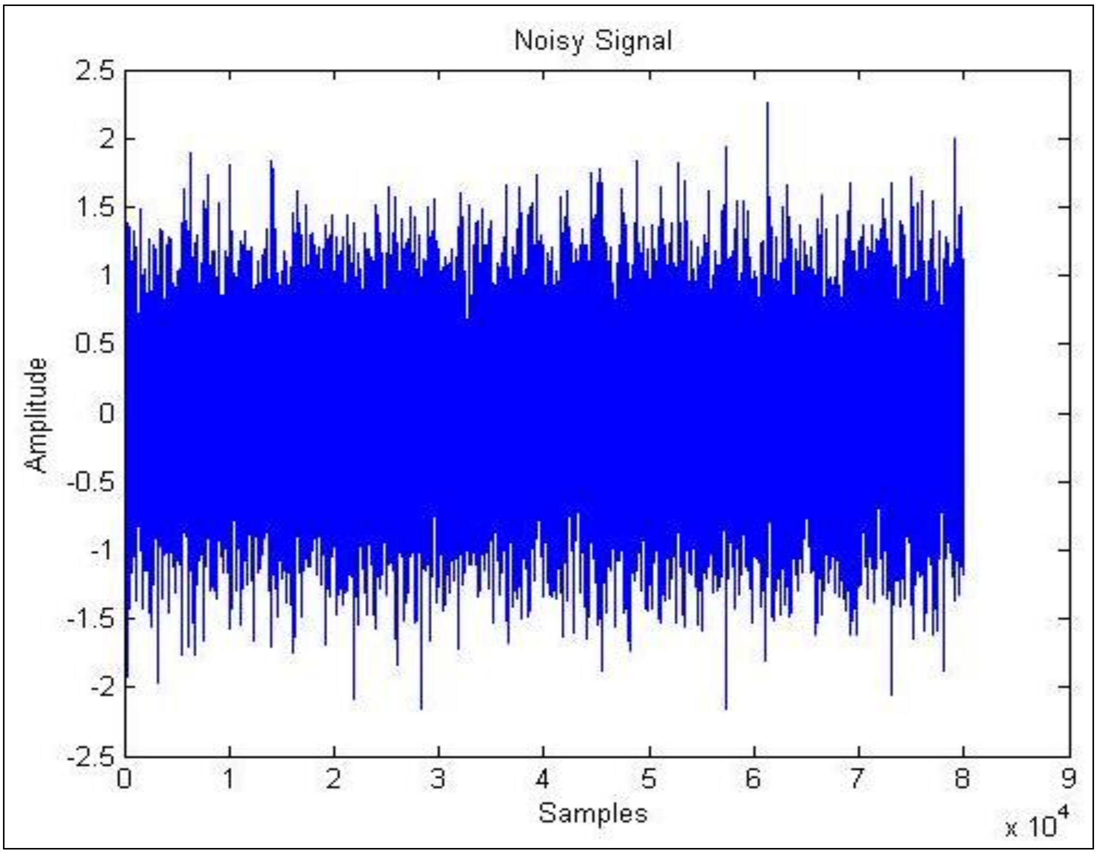

***Fig. 17.*** *Noisy Signal (Original Signal + Noise).*

 

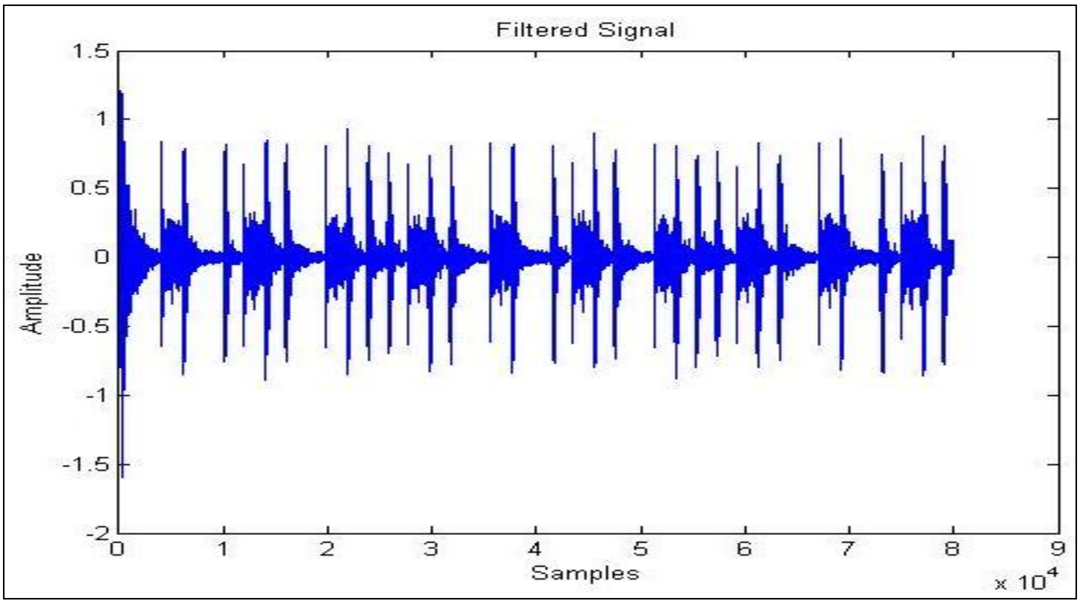


***Fig. 18.*** *Filtered Signal*.

Similarly in Simulink, the waterfall model output of down sampled filter weight is obtained which is shown in Figure 19. The waterfall block displays multiple vectors of data at one time. These vectors represent the input data at consecutive sample times. The input to the block can be real or complex-valued data vectors of any data type including fixed-point data types. The waterfall block displays only real-valued, double-precision vectors of data. Therefore, the input is converted to double-precision before the block processes the data. The data are displayed in a 3-dimensional axis in the waterfall window. By default, the x-axis represents amplitude, the y-axis represents samples, and the z-axis represents time.

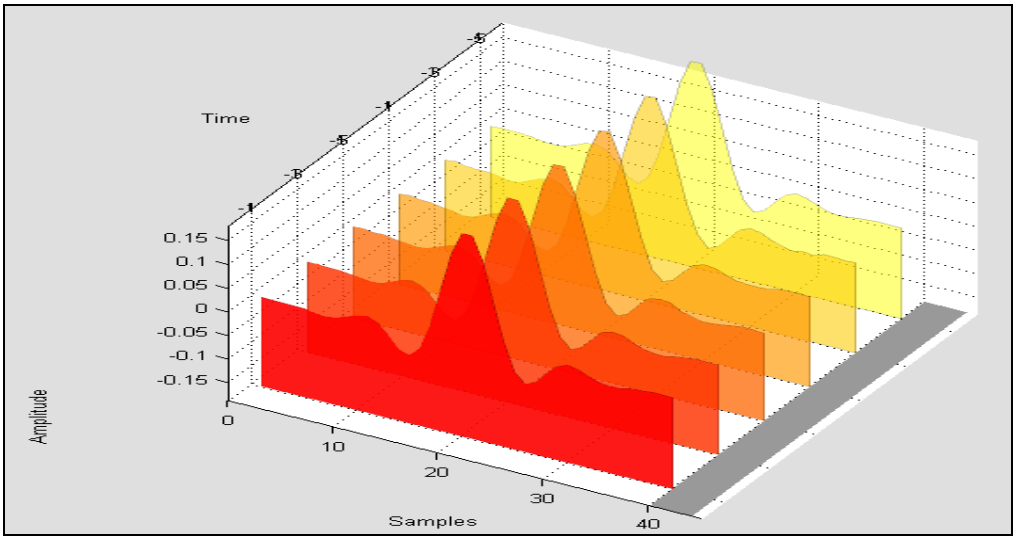


***Fig. 19.*** *Waterfall Model*.

## CONCLUSION

The NLMS algorithm is the most popular adaptive algorithm. It gives guaranteed stability and convergence. The principal advantage of the method is in its adaptive capability, its low output noise, and its low signal distortion. The adaptive capability allows the processing of inputs whose properties are unknown and in some cases non-stationary. Output noise and signal distortion are lower than can be achieved with conventional optimal filter configurations. The LMS algorithm adapts the filter tap weights so that e(n) is minimized in the mean-square sense.

## REFERENCES


1. Farhang-Boroujeny B. *Adaptive Filters: Theory and Applications*. John Wiley & Sons; 2013.
2. Lee K.A., Gan W.S., Kuo S.M. *Subband Adaptive Filtering: Theory and Implementation*. John Wiley & Sons; 2009.
3. Jiao Y., Cheung R.Y., Chow W.W., *et al*. “Signed-gradient adaptive step size LMS algorithm for biomedical applications”, *IEEE 36th Annual International Conference in Engineering in Medicine and Biology Society*. Embc, 2014, 3208–11p.
4. Marshall D.F., Jenkins W.K. “A fast quasi-newton adaptive filtering algorithm.” *IEEE Trans Signal Proc*. 1992; 40(7): 1652–62p.
5. Breining C., *et al*. “Acoustic echo control. An application of very-high-order adaptive filters”. *IEEE Signal Proc Mag.* 1999; 16(4): 42–69p.
6. Tüchler M., Singer A.C., Koetter R. “Minimum mean squared error equalization using a priori information”. *IEEE Trans Signal Proc*. 2002; 50(3): 673–83p.
7. Widrow B., *et al*. “Adaptive noise cancelling: principles and applications”. *Proc IEEE.* 1975; 63(12): 1692–1716p.
8. Chang C.Y., Pan S.T., Liao K.C. “Active noise control and its application to snore noise cancellation”. *Asian J Control* 2013; 15(6): 1648–54p.
9. Zhao S. *Performance Analysis and Enhancements of Adaptive Algorithms and Their Applications*, Doctoral Dissertation, Nanyang Technological University, 2009.
10. Harris R.W., Chabries D.M., Bishop F.A. “A variable step (vs) adaptive filter algorithm”. *IEEE Trans Acoustics, Speech Signal Process* 1986; 34(2): 309–16p.
11. Górriz J.M., Ramírez J., Cruces-Alvarez S., *et al*. “A novel LMS algorithm applied to adaptive noise cancellation”. *IEEE Signal Process Lett.* 2009; 16(1): 34–7p.
12. Lázaro-Gredilla M., Azpicueta-Ruiz L., Figueiras-Vidal A.R., *et al*. “Adaptively biasing the weights of adaptive filters”. *IEEE Trans Signal Process*. 2010; 58(7): 3890–5p.
13. Haykin S.S. *Adaptive Filter Theory*. India: Pearson Education India; 2008.
14. Bershad N.J. “Analysis of the normalized LMS algorithm with Gaussian inputs”. *IEEE Trans Acoustics, Speech Signal Process*. 1986; 34(4): 793–806p.