

# Performance Analysis of Adaptive Filters for Noise Cancellation in Audio Signal for Hearing Aid Application

G K Girisha<sup>1</sup>, S L Pinjare<sup>2</sup>

<sup>1</sup>Assistant Professor, NMIT, Bangalore, India

<sup>2</sup>Senior Professor, REVA University Bangalore, India

**Abstract:** This paper represents the experimental development of software and hardware configuration to implement adaptive algorithms for noise cancellation in audio signal processing for hearing aid applications. Adaptive FIR filters play an essential role in many DSP systems including noise cancellers, signal prediction, adaptive feedback cancellation and echo cancellation. In this paper, we describe adaptive filter which is used for the cancellation of the noise component which is overlapped with the desired signal in the same frequency range. The study includes comparison between three different types of adaptive filters, the LMS algorithm, NLMS algorithm and Recursive Least Squares Filter (RLS) algorithm. Mean Squared Error (MSE), Peak Signal to Noise Ratio (PSNR) and Mean Absolute Error (MAE) are calculated for all the three adaptive algorithms. The efficient Adaptive Noise Canceller is realized using MATLAB/Simulink.

**Keywords:** Adaptive filter algorithms, MSE, PSNR, MAE, microphone arrays.

## 1. Introduction

Microphone array consists of multiple microphones that are arranged in a specific pattern and can be used to capture the desired speech. The received microphone signals usually consist of a desired signal and noise. The received signals are processed (i.e. filtered and summed) in order to extract the desired signal. The noise signals are commonly composed of signals related to spatially coherent noise, spatially partly coherent noise and spatially incoherent noise. The spatially coherent noise is often used to model noise sources (e.g. interfering talkers) while the spatially partly coherent noise is used to model diffuse noise. For omnidirectional microphones, the spatial coherence of diffuse noise is almost one at low frequencies and almost zero at high frequencies.

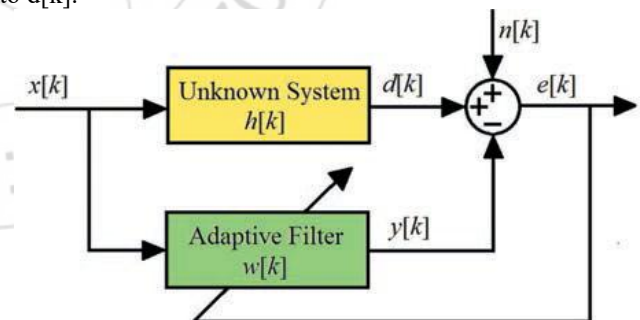
Adaptive filtering techniques have been successfully applied to communications systems such as smart antennas, channel equalization problems, interference cancellations, echo cancellation and spectral estimation for speech analysis and synthesis.

Adaptive filtering consists of two basic operations; the filtering process which generates an output signal from an input data signal, and the adaptation process which adjusts the filter coefficients in a way to minimize a desired cost function. Basically, there are a large number of filter structures and algorithms that have been used in adaptive filtering applications.

## 2. Adaptive Structure for System Identification:

Adaptive filters are an important part of signal processing. Adaptive filters adjust its transfer function according to an optimizing algorithm. Due to the complexity of the optimizing algorithms, most adaptive filters are digital

filters. Adaptive filters perform digital signal processing and adapt their performance based on the input signal. The aim to use an adaptive filter for system identification is to provide a linear model that represents the best fit to an unknown system, i.e. estimate the impulse response,  $h[k]$ , of the unknown system. Fig. 1 shows an adaptive filter structure that can be used for system identification or modeling. The input signal  $x[k]$  excites both the unknown system and the adaptive filter. The error signal  $e[k]$  is the difference between the unknown system response  $d[k]$  and the adaptive filter response  $y[k]$ . This error signal is fed back to the adaptive filter and is used to update the adaptive filter's coefficients until the overall output  $y[k]$  will be equal to  $d[k]$ .



**Figure 1:** Adaptive Filter Structure for System Identification

The purpose of the adaptive filter is adjusts its weights,  $w[k]$ , using the LMS algorithms, to produce an output  $y[k]$  that is as close as possible to the unknown system output  $d[k]$ . When this happens, the adaptation process is finished, and  $e[k]$  approaches zero. When MSE is minimized, the adaptive filter coefficients,  $w[k]$ , are approximately equal to the unknown system coefficients,  $h[k]$ . The internal plant noise is represented as an additive noise  $n[k]$ .

## 3. Adaptive LMS Algorithm

One of the most used algorithms for adaptive filtering is the

LMS algorithm. LMS adjusts the adaptive filter taps and modifying them by an amount proportional to the instantaneous estimate of the gradient of the error surface. It neither requires correlation function calculation nor matrix inversions, which makes it simple and easier when compared to other algorithm.

The adaptive LMS algorithm takes the following form:

$$w[k+1] = w[k] + 2 \mu e[k] x[k] \text{ Eq 1}$$

$$y[k] = w^T [k] x[k] \text{ Eq 2}$$

$$e[k] = d[k] - y[k] \text{ Eq 3}$$

Where indicates that the filter coefficient weight in the next state  $w[k+1]$  depends on the filter coefficient weight in its current state  $w[k] = [w_0[k] \ w_1[k] \ \dots \ w_N [k]]$ . The convergence factor  $0 < \mu < 1$  (referred to as step size), the error signal  $e[k]$ , the desired output  $d[k]$ , the filter output  $y[k]$  and input vector  $x[k] = [x[k] \ x[k-1] \ \dots \ x[k-N+1]]$ . The filter coefficients adjustment with this algorithm is performed until the MSE is minimized. The convergence factor selection  $\mu$  is essential, due it determines the convergence speed and the filter stability.

#### 4. Adaptive Normalized LMS Algorithm

This algorithm improve the convergence speed comparatively with the classical LMS algorithm, therefore, is more robust than the LMS algorithm. The NLMS algorithm employs the method of maximum slope, where the convergence factor presents a compromise between convergence speed and accuracy, i.e.  $\mu$  varies over time. The adaptive NLMS algorithm takes the following form:

$$w[k + 1] = w[k] + \mu \frac{x[k]}{\varepsilon + x^T[k]x[k]} e [k] \text{ Eq 4}$$

The parameters of this algorithm are the same of the LMS, in addition the term  $\varepsilon$  is a constant that prevents division by a very small number of data norm. This algorithm eliminates the strong dependence of data input, and the convergence algorithm depends directly of the input signal power to absorb large variations in the signal  $x[k]$ .  $e[k]$  is the error signal, obtained from the previous adaptive coefficients  $w[k-1]$ .

$$e[k] = d[k] - w^T [k] x [k] \text{ Eq 5}$$

In practice this amount is necessary because the weight cannot be updated until the arrival of the next sample.

$$K_N[k] = R^{-1}_N [k] x [k] \text{ Eq 6}$$

$$w[k+1] = w[k] + K_N [k] e [k] \text{ Eq 7}$$

$$w[k+1] = w[k] + K_N [k] (d [k] - y[k] ) \text{ Eq 8}$$

$$w[k+1] = w[k] + K_N [k] (d [k] - w^T [k] x [k] ) \text{ Eq 9}$$

The vector  $KN[k]$  is called Kalman gain and can be generated recursively without inverting the matrix  $R^{-1} N[k]$ . In this algorithm, the coefficients is updated for each sample at time  $k$ , this is done by taking into account the  $N$  previous entries.

#### 5. Recursive Least Squares Algorithm

The Recursive least squares (RLS) is an adaptive filter which recursively finds the coefficients that minimize a weighted linear least squares cost function relating to the input signals.

This is in contrast to other algorithms such as the least mean squares that aim to reduce the mean square error. In the derivation of the RLS, the input signals are considered deterministic, while for the LMS and similar algorithm they are considered stochastic. Compared to most of its competitors, the RLS exhibits extremely fast convergence. However, this benefit comes at the cost of high computational complexity.

#### 6. Simulation Methodology

This work used MATLAB Digital filter and Simulink environment for the filter realization. Various Parameters used for the simulation are filter Structure of filter length 32, adaptive algorithm (LMS, NLMS, RLS), Simulation time (10, 30, 50, 80), Design method (Kaiser Window  $B=0.5$ ) and frequency specifications (Normalized 0-1).

Figure 2 shows the MATLAB implementation of adaptive FIR Filter using NLMS algorithm. An audio file of frequency 44.1kHz, 16 bit is considered as a pure audio signal and is added with pink noise of frequency 44.1kHz, 16 bit. Later to remove the noise the NLMS adaptive filter is implemented. The LMS and RLS adaptive filter algorithms are implemented similarly.

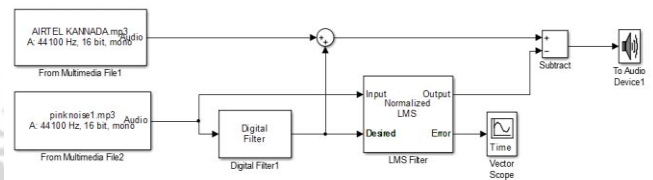


Figure 2 Adaptive filter implementation by using NLMS algorithm

Figure 3 represents the implementation of the algorithm to view the spectrum of all the signals.

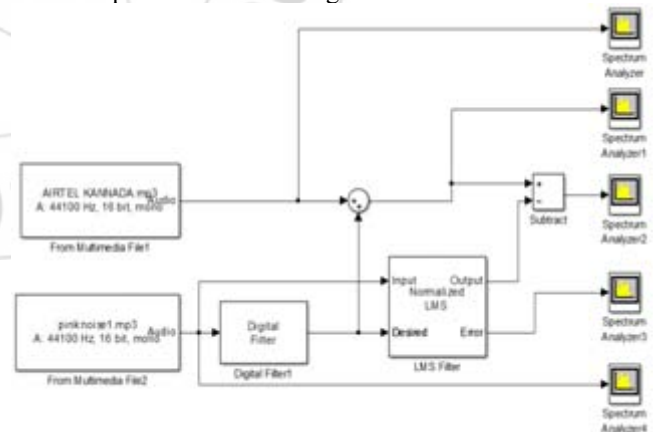
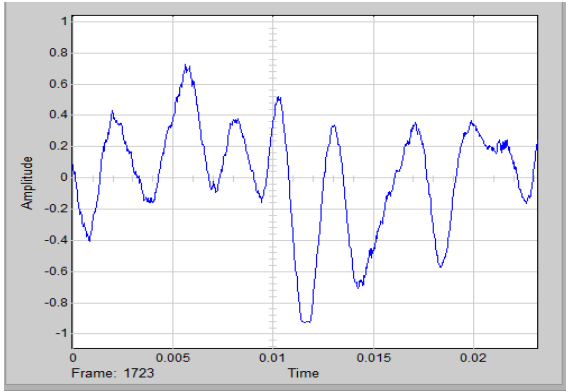


Figure 3: The adaptive filter implementation by using NLMS algorithm to view the spectrums

#### 7. Simulation Results

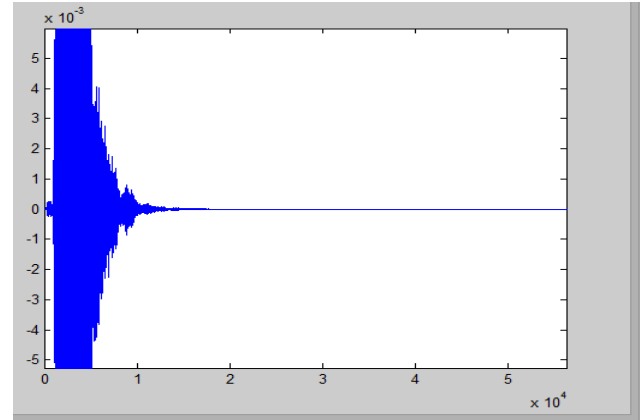
Results for the NLMS adaptive filter implementation are described below. The simulation time was 10sec.

Figures 4,5,6,7 shows time scope outputs of the input signal, noise signal, contaminated signal and the filtered signal. It is observed that noise from the signal is effectively removed.

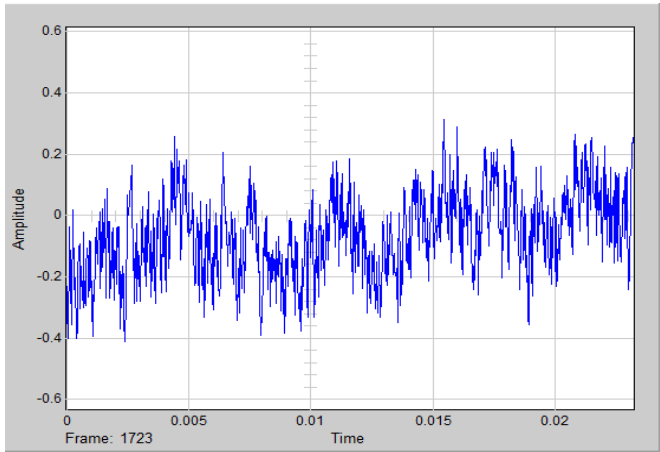


**Figure 4:** Audio speech signal (input) for adaptive filter.

From the figure 8 it has observed that, the error for the adaptive filter is approaching to zero.

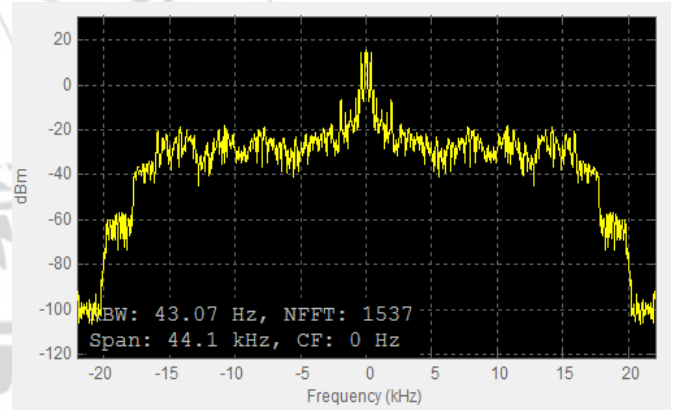


**Figure 8:** Adaptive filter error signal

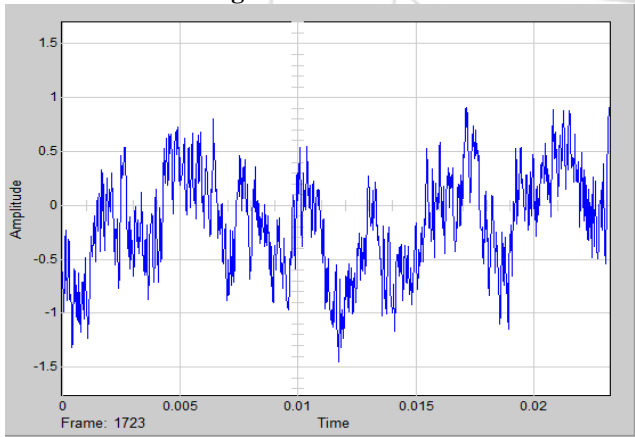


**Figure 5:** Pink Noise.

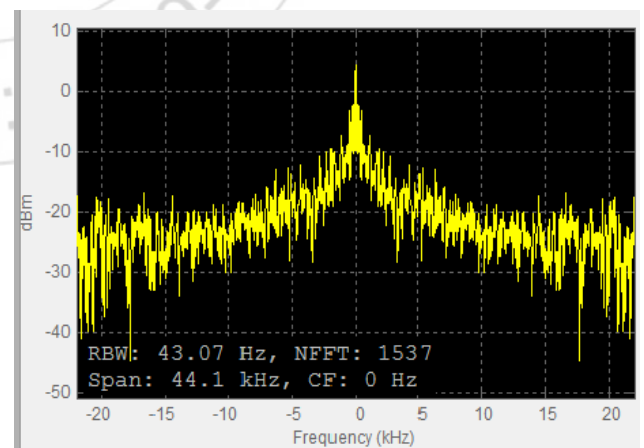
Figures 9,10,11,12 shows the spectrum analyzer output of adaptive FIR filter using NLMS algorithm for a simulation time of 10sec. It is observed in the spectrums that the noise from the signal is effectively removed.



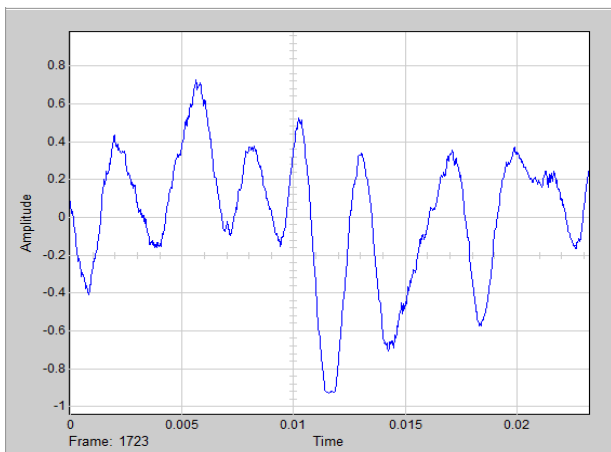
**Figure 10:** Spectrum of audio speech signal(input) for adaptive filter



**Figure 6:** Contaminated Signal (Pure signal added with noise)



**Figure 11:** Spectrum of pink noise signal



**Figure 7:** Output Signal after noise removal

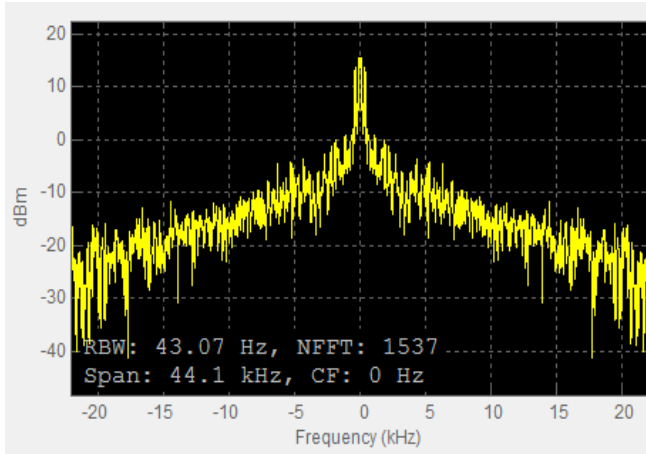


Figure 12: Spectrum of the contaminated signal

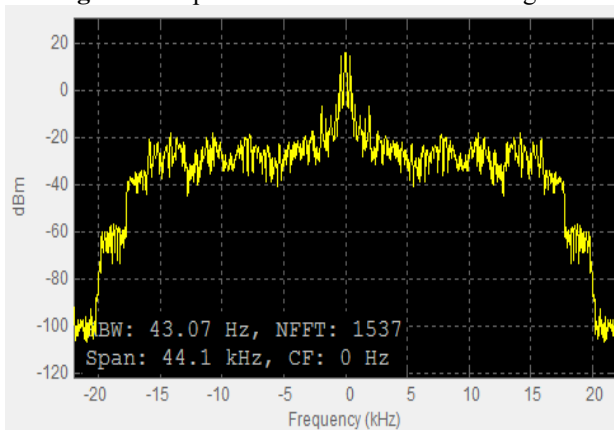


Figure 13: Spectrum of the output signal after adaptive filtering

## 8. Parameter Calculations

For the obtained results the following parameters are calculated to compare the performance among the filters being used.

### 8.1 Mean Squared Error (MSE)

The mean squared error (MSE) of an estimator measures the average of the squares of the errors, that is, the difference between the estimator and what is estimated. If  $\hat{Y}$  is a vector of  $n$  predictions, and  $Y$  is the vector of observed values corresponding to the inputs to the function which generated the predictions, then the MSE of the predictor can be estimated by

$$MSE = \frac{1}{n} \sum_{i=1}^n (\hat{Y}_i - Y_i)^2$$

### 8.2 Peak Signal to Noise Ratio (PSNR)

Peak signal-to-noise ratio, often abbreviated PSNR, is an engineering term for the ratio between the maximum possible power of a signal and the power of corrupting noise that affects the fidelity of its representation. PSNR is most commonly used to measure the quality of reconstruction of lossy compression codecs (e.g., for image compression). The signal in this case is the original data, and

the noise is the error introduced by compression. When comparing compression codecs, PSNR is an approximation to human perception of reconstruction quality. Although a higher PSNR generally indicates that the reconstruction is of higher quality, in some cases it may not.

The PSNR (in dB) is defined as:

$$\begin{aligned} PSNR &= 10 \cdot \log_{10} \left( \frac{MAX_I^2}{MSE} \right) \\ &= 20 \cdot \log_{10} \left( \frac{MAX_I}{\sqrt{MSE}} \right) \\ &= 20 \cdot \log_{10} (MAX_I) - 10 \cdot \log_{10} (MSE) \end{aligned}$$

### 8.3 Mean Absolute Error (MAE)

The mean absolute error (MAE) is a quantity used to measure how close forecasts or predictions are to the eventual outcomes. The mean absolute error is given by

$$MAE = \frac{1}{n} \sum_{i=1}^n |f_i - y_i| = \frac{1}{n} \sum_{i=1}^n |e_i|$$

As the name suggests, the mean absolute error is an average of the absolute errors, where  $f_i$  is the prediction and  $y_i$  the true value.

Values of the different parameters for the three adaptive filters are shown in Table 1.

Table 1: Parameter Calculations

	LMS	NLMS	RLS
MSE	5.482e-5	5.087e-7	1.42e-6
PSNR	138.94	159.26	154.797
MAE	5.37e-4	5.36e-5	6.15e-5

## 9. Conclusion

Adaptive FIR filters play an essential role in many DSP systems. The study included comparison between three different types of adaptive filters, the LMS algorithm, NLMS algorithm and Recursive Least Squares Filter (RLS) algorithm for noise cancellation in audio signals.

## References

- [1] T Meng and D Messerschmitt, "Arbitrarily high sampling rate adaptive filters", IEEE Transactions on Acoustics, Speech, and Signal Processing, Vol. 35 Issue. 4.
- [2] Mahesh S. Chavan, Nikos Mastorakis, "Studies on Implementation of Haar and Daubechies Wavelet for Denoising of Speech Signal", International Journal of Circuits, Systems and Signal Processing, pp. 83 -96 , Vol 4, 2010.
- [3] Ramesh.R and Nathia.R, "Realization of Fir Filter using Modified Distributed Arithmetic Architecture", Signal & Image Processing; An International Journal (SIPIJ) Vol.3, pp.83-94, No.1 2012.

- [4] P. P. Vaidyanathan and S. K. Mitra, "Low pass band sensitivity digital filters: A generalized viewpoint and synthesis procedures," Proc. IEEE, Apr.1984, pp. 404-423.
- [5] Markus Rupp , "A Family of Adaptive Filter Algorithms with Decorrelating Properties" , IEEE Transactions On Signal Processing, Vol. 46, No. 3, March 1998

