A Novel Approach of Acoustic Echo Cancellation Using Adaptive Filtering

P.Rajesh[1]
[1]*Asst. Professor ,*
Dept of E.I.E,
V.R.Siddhartha Engineering College,
Vijayawada, A.P

A.Sumalatha[2]
[2]*Asst. Professor ,*
Dept of E.I.E ,
V.R.Siddhartha Engineering College,
Vijayawada, A.P

ABSTRACT

Acoustic echo cancellation is a common occurrence in today's telecommunication systems. It occurs when an audio source and sink operate in full duplex mode .The signal interference caused by acoustic echo is distracting to both users and causes a reduction in the quality of the communication. This paper focuses on the use of adaptive filtering techniques to reduce this unwanted echo, thus increasing communication quality.

Adaptive filters alter their parameters in order to minimize a function of the difference between a desired target output and their output. In the case of acoustic echo in telecommunications, the optimal output is an echoed signal that accurately emulates the unwanted echo signal. This is then used to negate the echo in the return signal. The better the adaptive filter emulates this echo, the more successful the cancellation will be. This paper examines various techniques and algorithms of adaptive filtering, employing discrete signal processing in MATLAB. Also noise cancellation algorithms are implemented using simulink in MATLAB.

Keywords: *MSE(Mean Square Error),LMS,NLMS and RLS.*

1.Introduction

1.1 Need for Echo Cancellation

In this new age of global communications, wireless phones are regarded as essential communication tools and have a direct impact on peoples day to day personal and business communications. Subscriber demand for enhanced voice quality has driven a new and key technology termed echo cancellation, which can provide near wire line voice quality across a wire less network.

Today's subscriber use speech quality as a standard for assessing the overall quality of a network. So for improving voice quality of a call, the hybrid and acoustic echoes that are inherent within the communications infrastructure, should be removed effectively. Ultimately, the search for improved voice quality has led to intensive research into the area of echo cancellation. Such research is conducted with the aim of providing solutions that can reduce back ground noise and remove acoustic echoes. By employing echo cancellation technology, the quality of speech can be improved.

1.2 Basics of Echo

Echo is a phenomenon where a delayed and distorted version of an original signal is reflected back to the source. Echoes are heard as they are reflected from the floor, walls and other neighboring objects.If a reflected wave arrives after a very short time of direct sound, it is considered as a spectral distortion orreverberation. However, when the leading edge of the reflected wave arises a few tens of milliseconds after the direct sound, it is heard as a distinct echo. The most important factor in echoes is called end-to-end delay, which is also known as latency. It is the time between the generation of the sound at one end of the call and its reception at the other end. Round trip delay is the time taken to reflect an echo and is approximately twice that of end-to-end delay.

Echoes become annoying when the round trip delay exceeds 30ms.Such an echo is typically heard as a hollow sound. Echoes less than 30dB are unlikely to be noticed. However, when round-trip-delay exceeds 30ms and echo strength exceeds 30dB, echoes become steadily more disruptive.

1.3 Acoustic echo

The development of hands-free teleconferencing system gave rise to this echo. It is due

ISSN: 2278-0181
Vol. 1 Issue 5, July - 2012

to poor voice coupling between loud speaker and microphone in hand sets and hands-free devices.

This paper will focus on the occurrence of acoustic echo in telecommunication system. Such a system consists of coupled acoustic input and output devices, both of which are active concurrently. An example of this is hands-free telephony system. In this scenario the system has both an active loudspeaker and microphone input operating simultaneously. The system then acts as both a receiver and transmitter in full duplex mode. When a signal is received by the system, it is output through the loudspeaker into an acoustic environment. This signal is reverberated within the environment and returned to the system via the microphone input. These reverberated signals contain time delayed images of the original signal, which are then returned to the original sender (Figure 1 ak is the attenuation, tk is time delay). The occurrence of acoustic echo in speech transmission causes signal interference and reduced quality of communication.

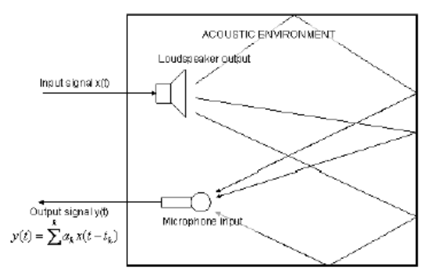

Fig :1 Origin of Acoustic Echo

1.4 Process of Echo Cancellation

An echo canceller is basically a device that detects and removes the echo of the signal from the far end after it has echoed on the local end's equipment.

The method used to cancel the echo signal is known as adaptive filtering. Adaptive filters are dynamic filters which iteratively alter their characteristics in order to achieve an optimal desired output. An adaptive filter algorithmically alters its parameters in order to minimize a function of the difference between the desired output d(n) and its actual output y(n). This function is known as the cost function of the adaptive algorithm. Figure 2 shows a block diagram of the adaptive echo cancellation system implemented throughout this paper. Here the filter H(n) represents the impulse response of the acoustic environment, W(n) represents the adaptive filter used to cancel the echo signal. The adaptive filter aims to equate its output y(n) to the desired output d(n) (the signal reverberated within the acoustic environment). At each iteration the error signal, e(n)=d(n)-y(n), is fed back into the filter, where the filter characteristics are altered accordingly.

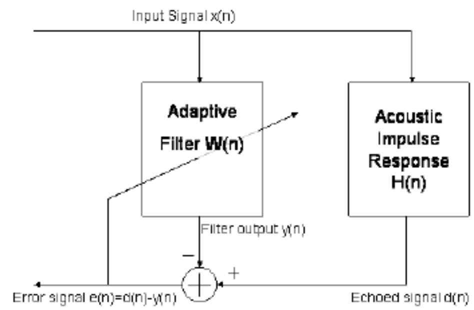

Fig:2 Block diagram of adaptive echo cancellation

2. Background Theory

In order to understand the content presented in this paper it is first necessary to provide some background information regarding digital signal theory. It will start out rather elementary and then progress to more complicated matters. Later we will build the theoretical basis in the derivation and implementation of the adaptive filtering techniques used in acoustic echo cancellation.

2.1 Discrete Time Signals

Real world signals, such as speech are analog and continuous. An audio signal, as heard by our ears is a continuous waveform which derives from air pressure variations fluctuating at frequencies which we interpret as sound. However, in modern day communication systems these signals are represented electronically by discrete numeric sequences. In these sequences, each value represents an instantaneous value of the continuous signal. These values are taken at regular time periods, known as the sampling period, Ts.

2.2 Transversal Fir Filters

A filter can be defined as a piece of software or hardware that takes an input signal and processes it so as to extract and output certain desired elements of that signal. There are numerous filtering methods, both analog and digital which are widely used. However, this paper shall be contained to adaptive filtering using a particular method known as transversal finite impulse response (FIR) filters.

The characteristics of a transversal FIR filter can be expressed as a vector consisting of values known as tap weights. It is these tap weights which determine the performance of the filter. These values are expressed in column vector form as, **w**(n) = [w0(n) w1(n) w2(n) .. wN-1(n)] ^{T}. This vector represents the impulse response of the FIR filter. The number of elements on this impulse response vector corresponds to the order of the filter, denoted in this project by the character N.

The utilization of an FIR filter is simple, the output of the FIR filter at time n is determined by the sum of the products between the tap weight vector, **w**(n) and N time delayed input values. If these time delayed inputs are expressed in vector form by the column vector **x**(n) = [x(n) x(n-1) x(n-2) .. x(n-N+1)]^T, the output of the filter at time n is expressed by equation 2.2. Throughout this paper the vector containing the time delayed input values at time n is referred to as the input vector, **x**(n). In adaptive filtering the tap weight values are time varying so for each at each time interval a new FIR tap weight vector must be calculated, this is denoted as the column vector.

w(n) = [w0(n) w1(n) w2(n) .. wN-1(n)] ^{T}.

$$y(n) = \sum_{i=0}^{N-1} w_i(n)x(n-i) \quad \text{(eq 2.2)}$$

It can be seen that this is also equivalent to the dot product between the impulse response vector and the input vector, and alternatively the product of the transpose of the filter tap weight vector and the input vector, see equation 2.3. Both of these equalities will be used throughout the paper.

$$y(n) = \mathbf{w}(n) \cdot \mathbf{x}(n)$$
$$y(n) = \mathbf{w}^T(n)\mathbf{x}(n) \quad \text{(eq 2.3)}$$

Figure 3 shows a block schematic of a real transversal FIR filter, here the input values are denoted by u(n), the filter order is denoted by M, and z^{-1} denotes a delay of one sample period.

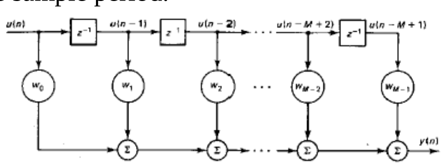

Fig:3 Transversal FIR filter

Adaptive filters utilize algorithms to iteratively alter the values of the impulse response vector in order to minimize a value known as the cost function. The cost function, ξ(n), is a function of the difference between a desired output and the actual output of the FIR filter. This difference is known as the estimation error of the adaptive filter, e(n) = d(n)-y(n). Section 3 will examine adaptive filter theory further and derive several algorithms used.

Like many signals in real world applications, the values of the input vector of the acoustic echo cancellation system are unknown before they arrive. Also as it is difficult to predict these values, they appear to behave randomly. So a brief examination of random signal theory will be treated in this section.

A random signal, expressed by random variable function, x(t), does not have a precise description of its waveform. It may, however, be possible to express these random processes by statistical or probabilistic models .single occurrence of a random variable appears to behave unpredictably. But if we take several occurrences of the variable, each denoted by n, then the random signal is expressed by two variables, x(t,n).

The main characteristic of a random signal treated in this paper is known as the expectation of a random signal. It is defined as the mean value across all n occurrences of that random variable, denoted by E[x(t)], where x(t) is the input random variable. It should be noted that the number of input occurrences into the acoustic echo cancellation system is always 1. Throughout this paper the expectation of an input signal is equal to the actual value of that signal. However, the E[x(n)] notation shall still be used in order to derive the various algorithms used in adaptive filtering.

2.4 Correlation Function

The correlation function is a measure of how statistically similar two functions are. The autocorrelation function of a random signal is defined as the expectation of a signals value at time n multiplied by its complex conjugate value at a different time m. This is shown in equation 2.4, for time arbitrary time instants, n and m.

$$\phi_{xx}(n,m) = E\left[x(n)x^*(m)\right] \quad \text{(eq 2.4)}$$

As this paper deals only with real signals the above equation becomes.

$$\phi_{xx}(n,m) = E\left[x(n)x(m)\right] \quad \text{(eq 2.5)}$$

The derivations of adaptive filtering algorithms utilize the autocorrelation matrix, **R**. For real signals this is defined as the matrix of expectations of the product of a vector **x**(n) and its transpose. This is shown in equation 2.6.

The autocorrelation matrix has the additional property that its trace, i.e. the sum of its diagonal elements, is equal to the sum of the powers of the values in the input vector

$$\mathbf{R} = \begin{bmatrix} E\left[\left|x_0(k)^2\right|\right] & E\left[x_0(k)x_1(k)\right] & \cdots & E\left[x_0(k)x_N(k)\right] \\ E\left[x_1(k)x_0(k)\right] & E\left[\left|x_1(k)^2\right|\right] & \cdots & E\left[x_1(k)x_N(k)\right] \\ \cdots & \cdots & \cdots & \cdots \\ E\left[x_N(k)x_0(k)\right] & E\left[x_N(k)x_1(k)\right] & \cdots & E\left[\left|x_N(k)^2\right|\right] \end{bmatrix}$$

$$\mathbf{R} = E\left[\mathbf{x}(k)\mathbf{x}^T(k)\right] \quad \text{(eq 2.6)}$$

As we will see later, sometimes a single value replaces one of the vectors in the
autocorrelation matrix, in this case the correlation function results in a vector. This vector is given by the expectation of that single value multiplied by the expectation of each of the values in the vector. Correlation matrices and vectors are based on either cross-correlation or autocorrelation functions. This simply refers to the signals being used in the function. If it is cross correlation, the signals are different, if it is autocorrelation, the two signals used in the function are the same.

2.5 Stationary Signals

A signal can be considered stationary in the wide sense, if the two following criteria are met.

1. The mean values, or expectations, of the signal are constant for any shift in time.

$$m_x(n) = m_x(n+k) \quad \text{(eq 2.7)}$$

2. The autocorrelation function is also constant over an arbitrary time shift.

$$\phi_{xx}(n,m) = \phi_{xx}(n+k, m+k) \quad \text{(eq 2.8)}$$

The above implies that the statistical properties of a stationary signal are constant over time. In the derivation of adaptive filtering algorithms it is often assumed that the signals input to the algorithm are stationary. Speech signals are not stationary in the wide sense, however do exhibit some temporary stationary behavior as will be seen in the next section.

2.6 Speech Signals

A speech signal consists of three classes of sounds. They are voiced, fricative and plosive sounds. Voiced sounds are caused by excitation of the vocal tract with quasi-periodic pulses of airflow. Fricative sounds are formed by constricting the vocal tract and passing air through it, causing turbulence that results in a noise-like sound. Plosive sounds are created by closing up the vocal tract, building up air behind it then suddenly releasing it, this is heard in the sound made by the letter .p.

Figure 4 shows a discrete time representation of a speech signal. By looking at it as a whole we can tell that it is non-stationary. That is, its mean values vary with time and cannot be predicted using the above mathematical models for random processes. However, a speech signal can be considered as a linear composite of the above three classes of sound, each of these sounds are stationary and remain fairly constant over intervals of the order of 30 to 40 ms.

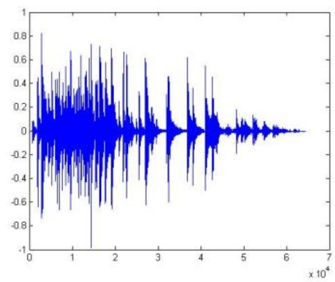

Fig: 4 Audio signal representation

The theory behind the derivations of many adaptive filtering algorithms usually requires the input signal to be stationary. Although speech is non-stationary for all time, it is an assumption of this paper that the short term stationary behaviour outlined above will prove adequate for the adaptive filters to function as desired.

2.7 Matrix Inversion Lemma

The matrix inversion lemma is an identity of matrix algebra, it is a simple way to determine the inverse of a matrix.

The matrix inversion lemma is as follows

Let **A** and **B** be two positive-definite MxM matrices, **C** be an MxN matrix and **D** is a positive-definite NxN matrix, (superscript H denotes hermitian transposition, which is transposition followed by complex conjugation)

$$\text{If } \mathbf{A} = \mathbf{B}^{-1} + \mathbf{C}\mathbf{D}^{-1}\mathbf{C}^H$$

$$\text{then } \mathbf{A}^{-1} = \mathbf{B} - \mathbf{B}\mathbf{C}(\mathbf{D} + \mathbf{C}^H\mathbf{B}\mathbf{C})^{-1}\mathbf{C}^H\mathbf{B} \quad \text{(eq 2.9)}$$

A special form of the matrix inversion lemma is used in the derivation of the recursive least squares (RLS) adaptive filtering algorithm in section 3.4. This special form is stated in equation 2.10, for an arbitrary non-singular NxN matrix **A**, any Nx1 vector **a,** and a scalar á
.

$$(\mathbf{A} + \alpha\mathbf{a}\mathbf{a}^T)^{-1} = \mathbf{A}^{-1} - \frac{\alpha\mathbf{A}^{-1}\mathbf{a}\mathbf{a}^T\mathbf{A}^{-1}}{1 + \alpha\mathbf{a}^T\mathbf{A}^{-1}\mathbf{a}} \quad \text{(eq 2.10)}$$

3. Echo Cancellation Algorithms

3.1 Basic Echo Canceller

A basic echo canceller used to remove echo in telecommunication network is presented in Figure 5

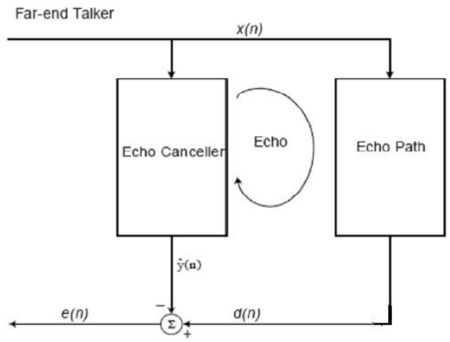

Fig:5 Basic Echo Canceller

The echo canceller mimics the transfer function of the echo path in order to synthesize a replica of the echo. Then the echo canceller subtracts the synthesized replica from the echo signal. However, the transfer function is unknown in practice. This problem can be solved by using an adaptive filter.

3.2 Adaptive Filter

The block diagram of adaptive filter is shown in figure 6

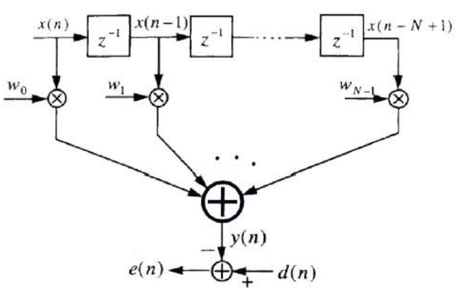

Fig:6 Block diagram of adaptive filter

It is made up of an echo estimator and subtractor. The echo estimator monitors the received path and dynamically builds a mathematical model of the line that creates the returning echo. The model of the line is convolved with the voice stream on the receive path. This yields an estimate of the echo, which is applied to the subtractor.The subtractor eliminates the linear part of the echo from the line in the send path. The echo canceller is said to converge on the echo as an estimate of the line is built through the adaptive filter.

Here w represents the coefficients of the FIR filter tap weight vector, x(n) is the input vector samples, z^{-1} is a delay of one sample periods, y(n) is the adaptive filter output, d(n) is the desired echoed signal and e(n) is the estimation error at time n. The aim of an adaptive filter is to calculate the difference between the desired signal and the adaptive filter output, e(n). This error signal is fed back into the adaptive filter and its coefficients are changed algorithmically in order to minimize a function of this difference, known as the cost function. In the case of acoustic echo cancellation, the optimal output of the adaptive filter is equal in value to the unwanted echoed signal. When the adaptive filter output is equal to desired signal the error signal goes to zero, in this situation the echoed signal would be completely cancelled and the far user would not hear any of their original speech returned to them. This section examines adaptive filters and various algorithms utilised. The various methods explained in this paper can be divided into two groups based on their cost functions. The first class are known as Mean Square Error (MSE) adaptive filters, they aim to minimize a cost function equal to the expectation of the square of the difference between the desired signal d(n), and the actual output of the adaptive filter y(n) (equation 3.1).

$$\xi(n) = E\left[e^2(n)\right] = E\left[(d(n) - y(n))^2\right] \quad \text{(eq 3.1)}$$

The second class are known as Recursive Least Squares (RLS) adaptive filters and they aim to minimize a cost function equal to the weighted sum of the squares of the difference between the desired and the actual output of the adaptive filter for different time instances. The cost function is recursive in the sense that unlike the MSE cost function, weighted previous values of the estimation error are also considered. The cost function is shown below in equation 3.2, the parameter λ is in the range of 0< λ <1. It is known as the forgetting factor as for λ <1 it causes the previous values to have an increasingly negligible effect on updating of the filter tap weights The value of 1/(1- λ) is a measure of the memory of the algorithm , this paper will primarily deal with infinite memory ,i.e. λ =1. The cost function for RLS algorithm, ξ(n) , is stated in equation..

$$\zeta(n) = \sum_{k=1} \rho_n(k) e_n^2(k)$$

$$\rho_n(k) = \lambda^{n-k} \quad \text{(eq 3.2)}$$

Where k=1, 2, 3….n., k=1 corresponds to the time at which the RLS algorithm

commences. Later we will see that in practice not all previous values are considered, rather only the previous N (corresponding to the filter order) error signals are considered. As stated previously, considering that the number of processes in our ensemble averages is equal to one, the expectation of an input or output value is equal to that actual value at a unique time instance. However, for the purposes of deriving these algorithms, the expectation notation shall still be used.

3.3 Wiener Filters

Wiener filters are a special class of transversal FIR filters which build upon the mean square error cost function of equation 3.1 to arrive at an optimal filter tap weight vector which reduces the MSE signal to a minimum. They will be used in the derivation of adaptive filtering algorithms in later sections

Consider the output of the transversal FIR filter as given below, for a filter tap weight vector, **w**(n), and input vector, **x**(n).

$$y(n)=\sum_{i=0}^{N-1} w(n)x(n-i)=\mathbf{w}^T(n)\mathbf{x}(n) \quad \text{(eq 3.3)}$$

The mean square error cost function can be expressed in terms of the cross-correlation vector between the desired and input signals, **p**(n)=E[**x**(n) d(n)], and the autocorrelation matrix of the input signal,R(n)=E[x(n)x^T(n)]

$$\begin{aligned}\xi(n) &= E\left[e^2(n)\right]\\ &= E\left[(d(n)-y(n))^2\right]\\ &= E\left[d^2(n)-2d(n)\mathbf{w}^T(n)\mathbf{x}(n)+\mathbf{w}^T(n)\mathbf{x}(n)\mathbf{x}^T(n)\mathbf{w}(n)\right]\\ &= E\left[d^2(n)\right]-2E\left[\mathbf{w}^T(n)\mathbf{x}(n)\right]+E\left[\mathbf{w}^T(n)\mathbf{x}(n)\mathbf{x}^T(n)\mathbf{w}(n)\right]\\ &= E\left[d^2(n)\right]-2\mathbf{w}^T\mathbf{p}+\mathbf{w}^T\mathbf{R}\mathbf{w}\end{aligned} \quad \text{(eq 3.4)}$$

When applied to FIR filtering the above cost function is an N-dimensional quadratic function. The minimum value of ξ(n) can be found by calculating its gradient vector related to the filter tap weights and equating it to 0.

for i = 0,1,...,N -1

$$\frac{\partial}{\partial w_i}=0 \text{ for } i=0,1,\ldots,\text{N - 1}$$

$$\nabla=\left[\frac{\partial}{\partial w_o}\frac{\partial}{\partial w_1}\cdots\frac{\partial}{\partial w_{N-1}}\right]^T$$

$$\nabla\xi=\mathbf{0} \quad \text{(eq 3.5)}$$

By finding the gradient of equation 3.4, equating it to zero and rearranging gives us the optimal wiener solution for the filter tap weights, **w**o.

$$\nabla\xi=\mathbf{0}$$
$$2\mathbf{R}\mathbf{w}_0-2\mathbf{p}=\mathbf{0}$$
$$\mathbf{w}_0=\mathbf{R}^{-1}\mathbf{p} \quad \text{(eq 3.6)}$$

The optimal wiener solution is the set of filter tap weights which reduce the cost function to zero. This vector can be found as the product of the inverse of the input vector autocorrelation matrix and the cross correlation vector between the desired signal and the input vector. The Least Mean Square algorithm of adaptive filtering attempts to find the optimal wiener solution using estimations based on instantaneous values.

3.4 LMS Algorithm

LMS algorithm was developed by Widrow and Hoff in 1959.It is widely used in various applications of adaptive filtering.The mail features that attracted the use of the LMS algorithm are low computational complexity, proof of convergence in stationary environments and stable behavior when implemented with finite precision arithmetic.

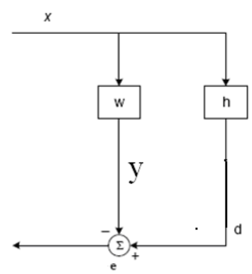

Fig:7 LMS algorithm

A path that changes the signal x is called h. Transfer function of this filter is not known in the beginning. The task of LMS algorithm is to estimate the transfer function of the filter.The result of the signal distortion is calculated by convolution and is denoted by d. In this case d is the echo and h is the transfer function of the hybrid. The adaptive algorithm tries to create a filter w. The transfer function in turn is used for calculating an estimate of the echo. The echo estimate is denoted by y.

The signals are added so that the output signal from the algorithm is

e=d-y

where e denotes the error signal.

The error signal and the input signal x are used for the estimation of the filter coefficient vector w. One of the main problems associated with choosing filter weight is that the path h is not stationary. Therefore, the filter weights must be updated frequently so that the adjustment to the variations can be performed. The filter is a FIR filter with the form

$$W= W_0(n)+W_1(n)Z^{-1}+\ldots+W_{N-1}(n)Z^{-(N-1)} \qquad \text{(eq.3.7)}$$

The LMS algorithm is a type of adaptive filter known as stochastic gradient based algorithm as it utilizes the gradient vector of the filter tap weights to converge on the optimal weiner solution.

With each iteration of the LMS algorithm , the filter tap weights of the adaptive filter are updated according to the following formula

$$W(n+1)=w(n)+2\mu e(n)x(n)$$

Here x(n) is the input vector of time delayed input values, $\mathbf{x}(n) = [x(n)\ x(n-1)\ x(n-2)\ \ldots x(n-N+1)]^T$ .The vector $\mathbf{w}(n) = [w0(n)\ w1(n)\ w2(n)\ ..\ wN-1(n)]^T$ represents the coefficients of the adaptive FIR filter tap weight vector at time n. The parameter μ is known as the step size parameter and is a small positive constant. This step size parameter controls the influence of the updating factor. Selection of a suitable value for μ is imperative to the performance of the LMS algorithm, if the value is too small the time the adaptive filter takes to converge on the optimal solution will be too long; if μ is too large the adaptive filter becomes unstable and its output diverges.

Implementation of the LMS algorithm.

Each iteration of the LMS algorithm requires 3 distinct steps in this order:

1.The output of the FIR filter, y(n) is calculated using equation

$$y(n) = \sum_{i=0}^{N-1} w(n)x(n-i) = \mathbf{w}^T(n)\mathbf{x}(n) \qquad \text{(eq 3.12)}$$

2.The value of the error estimation is calculated using equation 3.13.

$$e(n)=d(n)-y(n) \qquad \text{(eq 3.13)}$$

3. The tap weights of the FIR vector are updated in preparation for the next iteration, by equation 3.14.

$$\mathbf{w}(n+1) = \mathbf{w}(n) + 2\mu e(n)\mathbf{x}(n) \qquad \text{(eq 3.14)}$$

The main reason for the LMS algorithms popularity in adaptive filtering is its computational simplicity, making it easier to implement than all other commonly used adaptive algorithms. For each iteration the LMS algorithm requires 2N additions and 2N+1 multiplications (N for calculating the output, y(n), one for 2μe(n) and an additional N for the scalar by vector multiplication) .

3.5 NLMS algorithm

The NLMS algorithm utilizes a variable convergence factor, μ that minimizes the instantaneous error. Such a convergence factor usually reduces the convergence time but increases the mis adjustment.

One of the primary disadvantages of the LMS algorithm is having a fixed step size parameter for every iteration. This requires an understanding of the statistics of the input signal prior to commencing the adaptive filtering operation. In practice this is rarely achievable. Even if we assume the only signal to be input to the adaptive echo cancellation system is speech, there are still many factors such as signal input power and amplitude which will affect its performance.

The normalized least mean square algorithm (NLMS) is an extension of the LMS algorithm which bypasses this issue by selecting a different step size value, μ(n), for each iteration of the algorithm. This step size is proportional to the inverse of the total expected energy of the instantaneous values of the coefficients of the input vector **x**(n). This sum of the expected energies of the input samples is also equivalent to the dot product of the input vector with itself, and the trace of input vectors auto-correlation matrix, **R**.

$$tr[\mathbf{R}] = \sum_{i=0}^{N-1} E\left[x^2(n-i)\right] = E\left[\sum_{i=0}^{N-1} x^2(n-i)\right] \qquad \text{(eq 3.15)}$$

The recursion formula for the NLMS algorithm is stated in equation 3.16.

$$\mathbf{w}(n+1) = \mathbf{w}(n) + \frac{1}{\mathbf{x}^T(n)\mathbf{x}(n)} e(n)\mathbf{x}(n) \qquad \text{(eq 3.16)}$$

Implementation of the NLMS algorithm

The NLMS algorithm has been implemented in Matlab. As the step size parameter is chosen based on the current input values, the NLMS algorithm shows far greater stability with unknown signals. This combined with good convergence speed and relative computational simplicity make the NLMS algorithm ideal for the real time adaptive echo cancellation system. The code for the Matlab can be found in appendix A.

As the NLMS is an extension of the standard LMS algorithm, the NLMS algorithms practical implementation is very similar to that of the LMS algorithm. Each iteration of the NLMS algorithm requires these steps in the following order

1. The output of the adaptive filter is calculated.

$$y(n) = \sum_{i=0}^{N-1} w(n)x(n-i) = \mathbf{w}^T(n)\mathbf{x}(n) \quad \text{(eq 3.21)}$$

2. An error signal is calculated as the difference between the desired signal and the filter output.

$$e(n) = d(n) - y(n) \quad \text{(eq 3.22)}$$

3.The step size value for the input vector is calculated.

$$\mu(n) = \frac{1}{\mathbf{x}^T(n)\mathbf{x}(n)} \quad \text{(eq 3.23)}$$

4.The filter tap weights are updated in preparation for the next iteration.

$$\mathbf{w}(n+1) = \mathbf{w}(n) + \mu(n)e(n)\mathbf{x}(n) \quad \text{(eq 3.24)}$$

Each iteration of the NLMS algorithm requires 3N+1 multiplications, this is only N more than the standard LMS algorithm and as we shall see in section 4.2, this is an acceptable increase considering the gains in stability and echo attenuation achieved.

3.6 RLS algorithm

The other class of adaptive filtering techniques studied in this project is known as Recursive Least Squares (RLS) algorithms. These algorithms attempt to minimize the cost function in equation 3.33. Where k=1 is the time at which the RLS algorithm commences and λ is a small positive constant very close to, but smaller than 1. With values of λ<1 more importance is given to the most recent error estimates and thus the more recent input samples, this results in a scheme that places more emphasis on recent samples of observed data and tends to forget the past.

$$\zeta(n) = \sum_{k=1}^{n} \lambda^{n-k} e_n^2(k) \quad \text{(eq 3.33)}$$

Unlike the LMS algorithm and its derivatives, the RLS algorithm directly considers the values of previous error estimations. RLS algorithms are known for excellent performance when working in time varying environments. These advantages come with the cost of an increased computational complexity and some stability problems.

Implementation of the RLS algorithm

As stated the previously the memory of the RLS algorithm is confined to a finite number of values, corresponding to the order of the filter tap weight vector. Firstly, two factors of the RLS implementation should be noted: the first is that although matrix inversion is essential to the derivation of the RLS algorithm, no matrix inversion calculations are required for the implementation, thus greatly reducing the amount of computational complexity of the algorithm. Secondly, unlike the LMS based algorithms, current variables are updated within the iteration they are to be used, using values from the previous iteration.

To implement the RLS algorithm, the following steps are executed in the following order.

1.The filter output is calculated using the filter tap weights from the previous iteration and the current input vector.

$$\overline{y}_{n-1}(n) = \overline{\mathbf{w}}^T(n-1)\mathbf{x}(\mathbf{n}) \quad \text{(eq 3.41)}$$

2.The intermediate gain vector is calculated using

$$\mathbf{u}(n) = \tilde{\psi}_\lambda^{-1}(n-1)\mathbf{x}(n)$$

$$\mathbf{k}(n) = \frac{1}{\lambda + \mathbf{x}^T(n)\mathbf{u}(n)}\mathbf{u}(n) \quad \text{(eq 3.42)}$$

3.The estimation error value is calculated using equation 3.43.

$$\overline{e}_{n-1}(n) = d(n) - \overline{y}_{n-1}(n) \quad \text{(eq 3.43)}$$

4.The filter tap weight vector is updated using equation 3.43 and the gain vector
Calculated in equation 3.42.

$$\mathbf{w}(n) = \overline{\mathbf{w}}^T(n-1) + \mathbf{k}(n)\overline{e}_{n-1}(n) \quad \text{(eq 3.44)}$$

5. The inverse matrix is calculated using equation 3.45.

$$\psi_\lambda^{-1}(n) = \lambda^{-1}(\psi_\lambda^{-1}(n-1) - \mathbf{k}(n)[\mathbf{x}^T(n)\psi_\lambda^{-1}(n-1)]) \quad \text{(eq 3.45)}$$

Each iteration of the RLS algorithm requires $4N^2$ multiplication operations and $3N^2$ Additions. This makes its very costly to implement, thus LMS based algorithms, while they do not perform as well, are more favorable in practical situations.

4. Simulations Results

Each of the adaptive filtering algorithms were implemented using MATLAB.

4.1 LMS algorithm:

In this algorithm, the filter size is taken as 150, which is equal to the length of input signal. Step size parameter is taken as 0.8. Number of iterations are taken as 10000.

The figure below shows the input signal, desired signal, adaptive filter output signal and the error signal for the LMS algorithm.

Comparing this with LMS algorithm, the error signal is smaller in this algorithm. This increase in performance, as well as the increase in stability make it perfect for the real time echo cancellation system. Here the number of iterations are taken are 1000.

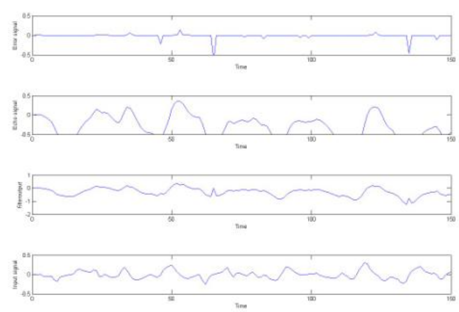

Fig:8 Input and Output waveforms of LMS algorithm

4.2 NLMS algorithm

The normalized LMS algorithm was simulated using Matlab, it is also the algorithm used in the real time echo cancellation system. Figure below shows the results of NLMS algorithm.

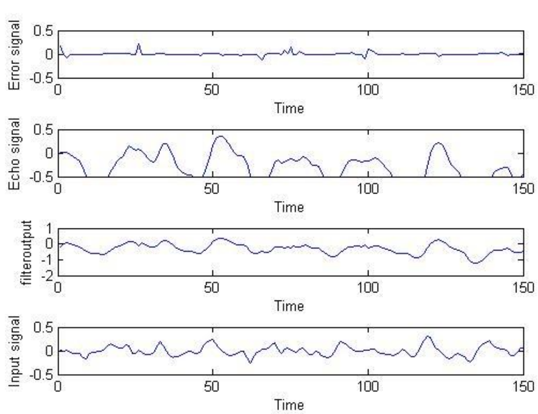

Fig:9 Input and Output waveforms of NLMS algorithm

4.3 RLS algorithm

The RLS algorithm was simulated using MATLAB. The figure below shows the results of RLS algorithm. In this the numbers of iterations used are 1000.

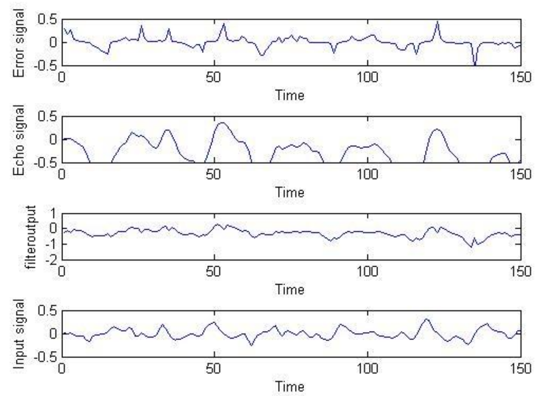

Fig:10 Input and Output waveforms of RLS algorithm

5. Conclusion and Futurescope

A summary of the performance of the adaptive filtering algorithms is expressed . It can be seen that when considering the attenuation values and the number of multiplication operations for each algorithm, the NLMS algorithm is the obvious choice for the real time acoustic echo cancellation system. Additionally, it does not require a prior knowledge of the signal values to ensure stability.

There are many possibilities for further development in this discipline, some of these are as follows.

The real time echo cancellation system can be implemented using the TI TMSC6711 DSK.

This paper dealt with transversal FIR adaptive filters, this is only one of many methods of digital filtering. Other techniques such as infinite impulse response (IIR) or lattice filtering may prove to be more effective in an echo cancellation application.

The algorithms studied in this paper perform best under purely stationary signal conditions. Strictly speaking speech does not fall into this category. Further work could be done in developing techniques specifically designed for non-stationary speech signals.

6.References

[1] Diniz, Paulo S. R. 1997, *Adaptive Filtering, Algorithms and Practical Implementation.* Kluwer Academic Publishers, Boston.
[2] Farhang-Boroujeny, B. 1999, *Adaptive Filters, Theory and Applications.* John Wiley and Sons, New York.
[3] Haykin, Simon. 1991, *Adaptive Filter Theory*. 2nd Edition. Prentice-Hall Inc., New Jersey.
[4] Homer, J. 1994, *Adaptive Echo Cancellation in Telecommunications*. PhD thesis,
[5] Oppenheim, Alan V., Schafer, Ronald W. 1989, *Distcrete-Time Signal Processing*, International edition. Prentice-Hall, Inc., New Jersey.

AUTHOR'S BIOGRAPHY

Mr.P.Rajesh[1] was born in Narasaraopet, Guntur district, A.P. On January 7, 1985.He received B.Tech degree in Electronics and Instrumentation Engineering branch from Bapatla Engineering College, Bapatla, A.P in 2007 and M.Tech degree in Instrumentation and Control systems Specialization from JNT University College of Engineering, JNTU Kakinada, Kakinada in 2010. He has 2 years of experience in teaching. He is at present working as a Assistant professor in V.R Siddhartha Engineering College, Vijayawada, A.P.

Mrs. A. Sumalatha[2] was born in Machilipatnam, Krishna Dt. A.P on August 25, 1980.She received her B.E degree in Electronics and Instrumentation Engineering from college of Engineering GITAM, Vizag in 2003 and M.Tech degree in Industrial Process Instrumentation specialization from college of engineering A.U in 2006, at present she is a research scholar in Dept. of Instrument Technology, A.U, Vizag. She has 6 years of experience in teaching, at present she is working as a Assistant professor in the department of Electronics and instrumentation, VR Siddhartha Engineering College, Vijayawada.