ACOUSTIC ECHO CANCELLATION USING CONVENTIONAL ADAPTIVE ALGORITHMS AND MODIFIED VARIABLE STEP SIZE LMS ALGORITHM

A
THESIS
Submitted in partial fulfillment of the requirements for the award of the degree of

MASTER OF ENGINEERING
In
ELECTRONICS AND COMMUNICATION ENGINEERING

Submitted By

Shobhna Gupta

Regn. No-8034130

Under the guidance of

Mr. Balwant Singh

Lecturer

DEPARTMENT OF ELECTRONICS AND COMMUNICATION ENGG.
THAPAR INSTITUTE OF ENGINEERING AND TECHNOLOGY
(DEEMED UNIVERSITY)
PATIALA –147004(PUNJAB)
INDIA

Certificate

I, Shobhna Gupta, hereby certify that the work presented in this thesis entitled **"Acoustic Echo Cancellation using Conventional Adaptive Algorithms and Modified Variable Step Size LMS algorithm"** by me in partial fulfillment of requirements for the award of degree of Master of Engineering (Electronics and Communication Engineering) at Thapar Institute of Engineering and Technology (Deemed University), Patiala, is an authentic record of my own work carried under the supervision of Mr. Balwant Singh.

The matter presented in this thesis has not been submitted in any other University / Institute for the award of any degree.

(Shobhna Gupta)

Signature of the Student

This is certified that the above statement made by the candidate is correct to the best of my knowledge.

(Balwant Singh)

Supervisor,

Department of Electronics and Communication Engineering,

Thapar Institute of Engineering and Technology,

Patiala-147004,

Punjab, India

(Dr.R.S.Kaler)

Professor and Head,

Dept of Elec. & Comm. Engineering,

Thapar Institute of Engineering and Technology,

Patiala-147004,

Punjab,India

(Dr. D.S.Bawa)

Dean of Academic Affairs

Thapar Institute Of Engg. &Tech.

Patiala-147004,

Punjab,India

Acknowledgement

This thesis is dedicated to my parents, who always supported me in doing the things my way and whose everlasting desire, selfless sacrifice, encouragement, affectionate blessings and help has made it possible for me to complete my degree.

I would like to take this opportunity to acknowledge the guidance and support given by my esteemed and worthy guide, **Mr. Balwant Singh**, Lecturer, Department of Electronics and communication Engg. His influence, leniency, vital suggestions, keen interest, inspiring guidance and approachability have in many ways assisted in my progress to the completion of this thesis.

I'd also like to acknowledge **Dr. R.S. Kaler**, Head of the Department, Electronics and communication Engg. , for his whole hearted assistance in official work and moral support given by him in every step of writing this thesis.

I am further indebted to **Mr. R.K. Khanna**, PG coordinator, for his scholarly guidance and wholehearted help during the course of study.

I wish to thank and express my gratitude to **Mr. Amit Kumar Bharwaj**, System Analyst cum Programmer, Impact Lab, for his timely help during lab work and **Mr. Jagdish Kumar** for his much needed help too.

I further like to say a big thanks to my brother, who has always been my best friend and advisor and all my friends who nudged me forward endlessly.

Above all I render my gratitude to the Almighty God who bestowed self-confidence, ability and strength in time to complete this task.

Place : T.I.E.T., Patiala **(Shobhna Gupta)**

Abstract

Adaptive filtering constitutes one of the core technologies in digital signal processing and finds numerous application areas in science as well as in industry. Adaptive filtering techniques are used in a wide range of applications, including echo cancellation, adaptive equalization, adaptive noise cancellation, and adaptive beamforming.

Acoustic echo cancellation is a common occurrence in todays telecommunication systems. The signal interference caused by acoustic echo is distracting to users and causes a reduction in the quality of the communication. Echo cancellers are very successful and today almost no echo at all can be perceived while using telephones

A variable step size least mean square algorithm (VSS LMS) is given with significant changes in which scalar step size increases or decreases as the squared error increases or decreases, thereby allowing the adaptive filter to track changes in the system and produces a smaller steady state error. A new VSS LMS algorithm is proposed, which can effectively adjust the step size while maintaining the immunity against independent noise disturbance. The modified variable step size LMS (MVSS LMS) algorithm allows more flexible control of misadjustment and convergence time without the need to compromise one for the other. The present work focuses on the conventional adaptive algorithms and modified variable step size LMS algorithm to reduce the unwanted echo, thus improving communication quality.

Simulation results are presented to support the analysis and to compare the performance of the modified algorithm with the other conventional adaptive algorithms. They show that the MVSS algorithm provides faster speed of convergence among other variable step size algorithms while retaining the same small level of misadjustment and the mean square error is almost similar to that of the (fixed step size least mean square) FSS-LMS algorithm under similar conditions. An attempt has been made to examine adaptive filtering techniques as they apply to acoustic echo cancellation, to simulate these adaptive filtering algorithms using MATLAB and to compare the performance of these adaptive filtering algorithms as they are applied to the acoustic echo cancellation application.

List of Figures

Fig: 1.1 System Identification Model

Fig: 1.2 Noise Cancellation Model

Fig: 1.3 Predicting Future Values of A Periodic Signal

Fig: 1.4 Interference Cancellation Model

Fig: 1.5 A Baseband Communication System

Fig: 1.6 A Simple Channel Equalizer Configuration

Fig: 1.7 Major Components of an Echo Canceller

Fig: 1.8 Echo Canceller Configuration

Fig 1.9: Block Diagram of an Adaptive Echo Cancellation System

Fig: 2.1 Prototype Wiener Filtering Scheme

Fig 2.2 Structure of an Adaptive Transversal Filter

Fig 3.1 The Grouping of LMS Offspring Algorithms

Fig 3.2 Learning Curve of LMS Algorithm

Fig 4.1 Comparison of Excess MSE of Various Adaptive Algorithms For White Input Case SNR = 0db

Fig 5.1 Origin of Acoustic Echo

Fig 5.2 Hands Free Telephony

Fig 5.3 Configuration of an Acoustic Echo Canceller

Fig 5.4 Direct Form II IIR Filter Realization For M=N

Fig 5.5 Direct Form FIR Filter Realization

Fig 5.6 Simulation Results For Echo Cancellation Using LMS Algorithm (Number Of Iterations = 25000 And Filter Order 1000)

Fig 5.7 Simulation Results For Echo Cancellation Using NLMS Algorithm (Number Of Iterations = 25000 And Filter Order 1000)

Fig 5.8 Comparison of the ERLE Plots for the LMS and NLMS Algorithm for 7500 Iterations

Fig 5.9 Comparison of the ERLE Plots for the LMS and NLMS Algorithm for 25000 Iterations

Fig 5.10 Simulation Results For Echo Cancellation Using RLS Algorithm (Number Of Iterations = 25000 And Filter Order 1000)

Fig 5.11 Comparison of Convergence Rate of Conventional Algorithms (LMS, NLMS and RLS Algorithms)

Fig 5.12 Simulation Results For Echo Cancellation Using Variable Step Size LMS Algorithm (Number Of Iterations = 25000 And Filter Order 1000)

Fig 5.13 Comparison of the Misadjustments Using the MVSS and VSS

Fig 5.14 Simulation Results for Echo Cancellation Using Modified Variable Step Size LMS Algorithm (No. Of Iterations = 25000 And Filter Order 1000)

Fig 5.15 Simulation Results For Echo Cancellation Using Gradient Adaptive Step Size LMS Algorithm (No. Of Iterations = 25000 And Filter Order 1000)

List of Abbreviations

AWGN	Additive White Gaussian Noise
AEC	Acoustic Echo Cancellation
B	Channel Bandwidth
C	Channel Capacity
CCI	Co-Channel Interference
DFE	Decision Feedback Equalizer
DSP	Digital Signal Processing
ERLE	Echo Return Loss Enhancement
ERL	Echo Return Loss
EC	Echo Canceller
FIR	Finite Impulse Response
FSS-LMS	Fixed Step Size Least Mean Square
GLMS	Gradient Adaptive Step Size Least Mean Square
ISI	Intersymbol Interference
IIR	Infinite Impulse Response
LMF	Least Mean Forth
LMS	Least Mean Square
LTE	Linear Transversal Equalizer
MSE	Mean Square Error
MASE	Mean Asymptotic Square Error
MA	Moving Average
MVSS	Modified Variable Step Size
NLMS	Normalized Least Mean Square
RLS	Recursive Least Squares
SNR	Signal to Noise Ratio
SE	Sign Error
SR	Sign Regressor
VOP	Voice Over Packet
VSS-LMS	Variable Step Size

Contents

Certificate

Acknowledgement

Abstract

List of Figures

List of Abbreviations

Chapter 1

Introduction

1.1 Introduction

1.2 Adaptive Digital Signal Processing

1.3 Applications of Adaptive Filters

1.3.1 System Identification

1.3.2 Noise Cancellation in Speech Signals

1.3.3 Signal Prediction

1.3.4 Interference Cancellation

1.3.5 Channel Equalization

1.3.6 Echo Cancellation

1.3.6.1 Echo Canceller Configuration

1.4 Adaptive Filtering Algorithms

1.4.1 VSS-LMS Algorithm

1.4.2 Modified VSS-LMS Algorithm

1.4.3 Recursive Least Squares (RLS) Algorithm

1.5 Performance of An Adaptive Algorithm

1.6 Adaptive Echo Cancellation

1.7Layout of Thesis

Chapter 2

Adaptive Filter Theory

2.1Introduction

2.2 The Wiener Filter

2.2.1 Mean Square Error Criterion

2.3 Wiener Filter: Transversal, Real Valued Case

2.4 Iterative Search Algorithm

2.5 Method of Steepest Descent

2.6 Error Performance Surface

Chapter 3

Adaptive Filtering Algorithms

3.1 Introduction

3.2 Adaptive Filtering Algorithms

3.2.1 LMS Algorithm

3.2.2 Derivation of LMS Algorithm

3.2.3 Step Size Optimization

3.2.4 Effect of Power Spectral Density Of Input Signal

3.2.5 Effect of Filter Tap Length

3.2.6 Effect of Input Signal Power

3.2.7 Normalized LMS Algorithm

3.2.8 RLS Algorithm

3.2.9 Derivation of RLS Algorithm

Chapter 4

Modified Variable Step Size LMS Algorithm

4.1 Introduction

4.2 Variable Step Size Algorithms

4.3 Modified Variable Step Size LMS Algorithm

4.3.1 Algorithm Formulation

4.3.2 Analysis of Modified VSS LMS Algorithm

4.3.3 Simulation Results and Observations

4.4 Gradient Adaptive Step Size LMS Algorithm

Chapter 5

Echo Cancellation-Analysis and Observations using Various Adaptive Algorithms

5.1 Introduction

5.2 Acoustic Echo Cancellation

5.2.1 Origin of Acoustic Echo

5.2.2 Hands Free Telephony

5.2.3 The Acoustic Echo Cancellation Problem

5.2.4 Configuration of An Acoustic Echo Canceller

5.3 Filter Types for Acoustic Echo Cancellation

5.3.1 IIR Filter

5.3.2 The IIR Filter in Acoustic Echo Cancellation

5.3.3 FIR Filter

5.3.4 The FIR Filter in Acoustic Echo Cancellation

5.4 Echo Cancellation Using LMS Algorithm

5.4.1 Simulation Results for Echo Cancellation Using LMS Algorithm

5.5 Echo Cancellation Using NLMS Algorithm

5.5.1 Simulation Results for Echo Cancellation Using NLMS Algorithm

5.6 Echo Cancellation Using RLS Algorithm

5.6.1 Simulation Results for Echo Cancellation Using RLS Algorithm

5.7 Comparison of Convergence Rate of Various Algorithms

5.8 Echo Cancellation Using VSS LMS Algorithm

5.8.1 Simulation Results For Echo Cancellation Using VSS LMS Algorithm

5.9 Echo Cancellation Using Modified VSS LMS Algorithm

5.9.1 Simulation Results for Echo Cancellation Using MVSS LMS Algorithm

5.10 Echo Cancellation Using Gradient Adaptive Step Size LMS Algorithm

5.10.1 Simulation Results for Echo Cancellation Using GLMS Algorithm

Chapter 6

Conclusions and Future Scope

6.1 Conclusions

6.2 Future Scope

Appendix A

Appendix B

References

Chapter 1

INTRODUCTION

1.1 Introduction

Signal processing is concerned with the representation, transformation and manipulation of signals and the information they contain. Prior to 1960's the technology used for signal processing was almost exclusively continuous time analog technology. The rapid advancement in the digital computer technology and integrated circuit fabrication resulted in an area of science and engineering called Digital signal Processing. It is because of the programming capability, low cost, miniature size, and low power consumption that widespread application of DSP techniques is being carried out [1]. The adaptive signal processing is a specialized branch of digital signal processing, which deals with adaptive filters and their applications.

1.2 Adaptive Digital Signal Processing

Filters are electric networks which permit unattenuated transmission of electric signals with in a specified frequency range and simultaneously suppress the signals outside this range. Filtering means the extraction of information about a quantity of interest at time 't' by using data measured up to and including time t. If the inputs to the filter are stationary, the resulting solution of the filtering problem is known as the Wiener Filter, which is said to be optimum in mean square sense. But it requires a priori information about the statistics of the data to be processed. If the environment is unknown then another efficient method is to use an adaptive filter using recursive algorithm. The algorithm starts with some predetermined set of initial conditions, representing whatever we know about the environment. In stationary environment it is expected that the adaptive filter converge to the optimum Wiener solution after successive iterations of the algorithm. In non- stationary environment, the algorithm can track time variations in the statistics of input data, provided that the variations are sufficiently slow.

Adaptive filtering constitutes one of the core technologies in digital signal processing and finds numerous application areas in science as well as in industry. Adaptive

filtering techniques are used in a wide range of applications, including echo cancellation, adaptive equalization, adaptive noise cancellation, and adaptive beamforming. These applications involve processing of signals that are generated by systems whose characteristics are not known a priori. Under this condition, a significant improvement in performance can be achieved by using adaptive rather than fixed filters. An adaptive filter is a self-designing filter that uses a recursive algorithm (known as adaptation algorithm or adaptive filtering algorithm) to "design itself." The algorithm starts from an initial guess; chosen based on the a priori knowledge available to the system, then refines the guess in successive iterations, and converges, eventually, to the optimal Wiener solution in some statistical sense [2]. In most of the practical applications where adaptive filters are used to estimate unknown system responses, one obviously desires to perform this estimation as accurately and quickly as possible.

The one basic common feature of adaptive filters is:

An input vector and a desired response are used to compute an estimation error, which in turn is used to control the values of a set of adjustable filter coefficients by a feedback loop and an algorithm.

1.3 Applications of Adaptive Filters

1.3.1 System Identification

System identification refers to the ability of an adaptive system to find the FIR filter that best reproduces the response of another system, whose frequency response is a priori unknown. The set up is shown in fig 1.1:

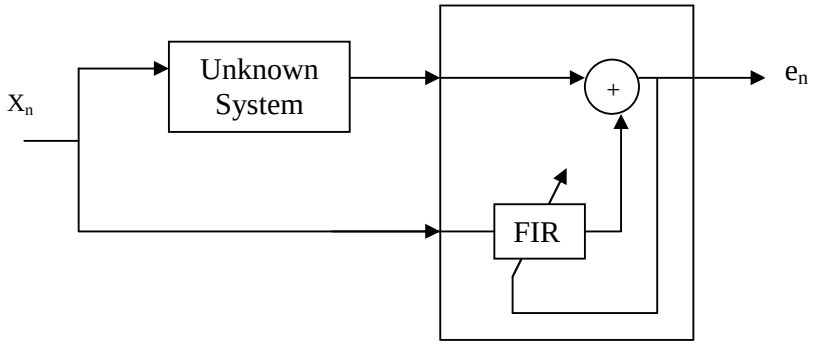


Fig: 1.1 System Identification Model

When the adaptive system reaches its optimum value and the output (*e*) is zero an FIR filter is obtained whose weights are the result of the adaptation process that is giving the same output as that of the 'unknown system' for the same input. In other words, the FIR filter reproduces the behavior of the 'unknown system'. This works perfectly when the system to be identified has got a frequency response that matches with that of a certain FIR filter. But if the unknown system is an all-pole filter, then the FIR filter will try its best. It will never be able to give zero output but it may reduce it by converging to an optimum weights vector. The frequency response of the FIR filter will not be exactly equal to that of the 'unknown system' but it will certainly be the best approximation to it.

1.3.2 Noise cancellation in speech signals

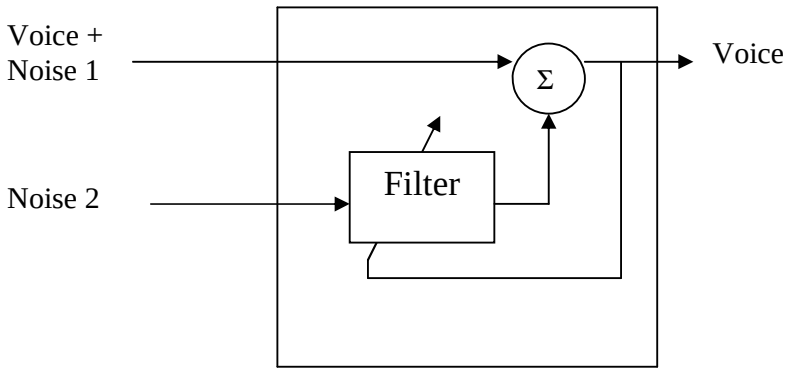


Fig: 1.2 Noise Cancellation Model

Adaptive filtering can be extremely useful in cases where a speech signal is submerged in a very noisy environment with many periodic components lying in the same bandwidth as that of speech. The adaptive noise canceller for speech signals needs two inputs. The main input is containing the voice that is corrupted by noise. The other input (noise reference input) contains noise related in some way to that of the main input (background noise). The system filters the noise reference signal to make it more similar to that of the main input and that filtered version is subtracted from the main input. Ideally it removes the noise and leaves intact the speech. In

practical systems noise is not completely removed but its level is reduced considerably.

1.3.3 Signal Prediction

Predicting signals may seem to be an impossible task, without some limiting assumptions. Assume that the signal is either steady or slowly varying over time, and periodic over time as well. Here the function of the adaptive filter is to provide best prediction (in some sense) of the present value of a random signal.

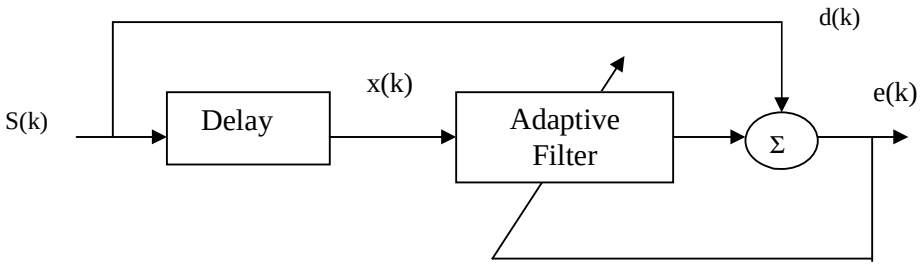


Fig: 1.3 Predicting Future Values of a Periodic Signal

Accepting these assumptions, the adaptive filter must predict the future values of the desired signal based on past values. When s(k) is periodic and the filter is long enough to remember previous values, this structure with the delay in the input signal, can perform the prediction. This structure can also be used to remove a periodic signal from stochastic noise signals. The present value of the signal serves the purpose of a delayed response for the adaptive filter. Past values of the signal supply the input applied to the adaptive filter. Depending upon the application of interest, the adaptive filter output or the estimation (prediction) error may serve as the system output. In the first case, system operates as a predictor, in the latter case; it operates as a prediction error filter.

Finally, it can be noticed that most systems of interest contain elements of more than one of the four adaptive filter structures. Carefully reviewing the real structure may be required to determine what the adaptive filter is adapting to. Also, for clarity in the figures, the analog-to-digital (A/D) and digital-to-analog (D/A) components do not

appear. Since the adaptive filters are assumed to be digital in nature and many of the problems producing analog data, converting the input signals to and from the analog domain is probably necessary. These models are used in the design of adaptive filters and those are further used in various applications of advanced communication.

1.3.4 Interference Cancellation

In this application, adaptive filter is used to cancel unknown interference contained alongside an information signal component in a primary signal, with the cancellation being optimized in some sense in fig 1.4. The primary signal serves as the desired response for the adaptive filter. A reference (auxiliary) signal is employed as the input to the adaptive filter. The reference signal is derived from the sensor or set of sensors located in relation to the sensors supplying the primary signal in such a way that the information signal component is weak or essentially undetectable.

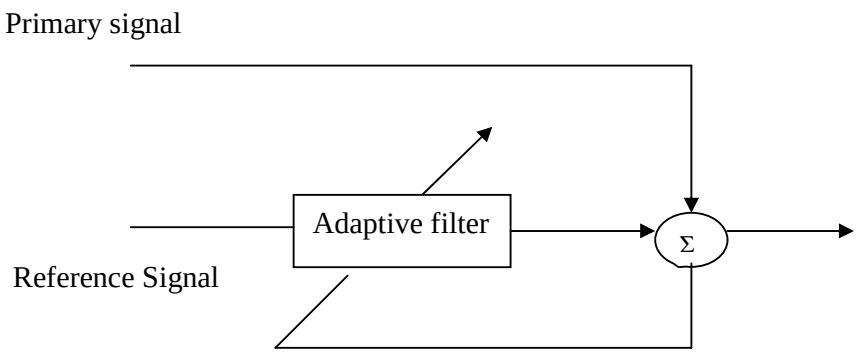


Fig: 1.4 Interference Cancellation Model

1.3.5 Channel Equalization

Communication channels such as telephone, wireless and optical channels are susceptible to intersymbol interference (ISI). Without channel equalization, the utilization of the channel bandwidth becomes inefficient. Channel equalization is a process of compensating for the effects caused by a band-limited channel, hence enabling higher data rates [3]. These disruptive effects are due to the dispersive transmission medium (e.g. telephone cables) and the multipath effects in the radio channel. A typical communication system is depicted in fig 1.5, where the equalizer is incorporated within the receiver while the channel introduces intersymbol

interference. The transfer function of the equalizer is an estimate of the direct inverse of the channel transfer function. To transmit high-speed data over a bandlimited channel, the frequency response of the channel is usually not known with sufficient precision to design an optimum match filter. The equalizer is, therefore, designed to be adaptive to the channel variation. The configuration of an adaptive equalizer is depicted in fig 1.6. Based on the observed channel output, an adaptive algorithm recursively updates the equalizer to reconstruct the output signal.

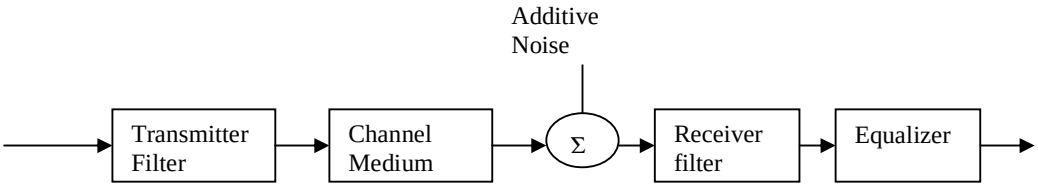


Fig: 1.5 A Baseband Communication System

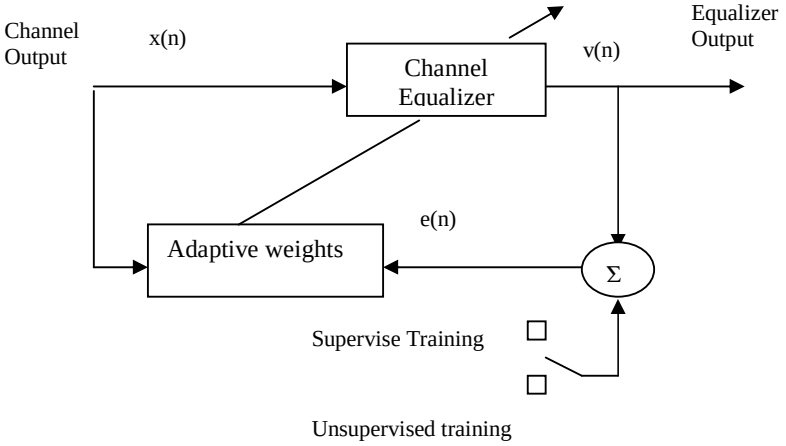


Fig1.6 A Simple Channel Equalizer Configuration

1.3.6 Echo Cancellation

Echo cancellation is one of the most widely used digital signal processing devices in the world because each telephone call requires a pair of echo cancellers. Basically, a transversal filter, which is adaptively modeling the echo path impulse responses, generates an estimate of the echo, with this an echo estimate is created at the right time to cancel the actual echo [4]. The common problems faced by echo cancellation are the convergence time and the degree of cancellation. Convergence time is the time

taken to reach an acceptable level of steady state residual echo. Degree of cancellation is the amount of echo cancelled, measured in Echo Return loss Enhancement [5] (ERLE).

Echo cancellation consists of three important blocks:

1. **Non-linear Processor**, to remove uncancelled residual echoes.

2. **Adaptive Echo canceller**, to adapt the echo path impulse responses and synthesize replica echoes.

3**. Double-talk Detector**, to detect double talk and consequently stop the echo canceller adaptation.

The echo cancellation block diagram is shown in fig 1.7**.** The adaptive echo canceller is the engine of echo cancellation, with its performance being mainly determined by the adaptive algorithm. The Normalized least mean square (NLMS) algorithm is the most popular algorithm implemented in echo cancellation. It is a well-studied algorithm, simple to implement and it guarantees convergence [6]. The non-linear processor is used to remove residual echoes by injecting comfort noise (also called pink noise). A low level of comfort noise is usually added during the silence periods as an audio-psychological comfort effect for the listener. The double-talk detector is used to sense the near-end and far end signal levels. Once both activities are detected double talk is declared and the adaptation of the echo canceller is stopped until the system is no longer in double talk mode.

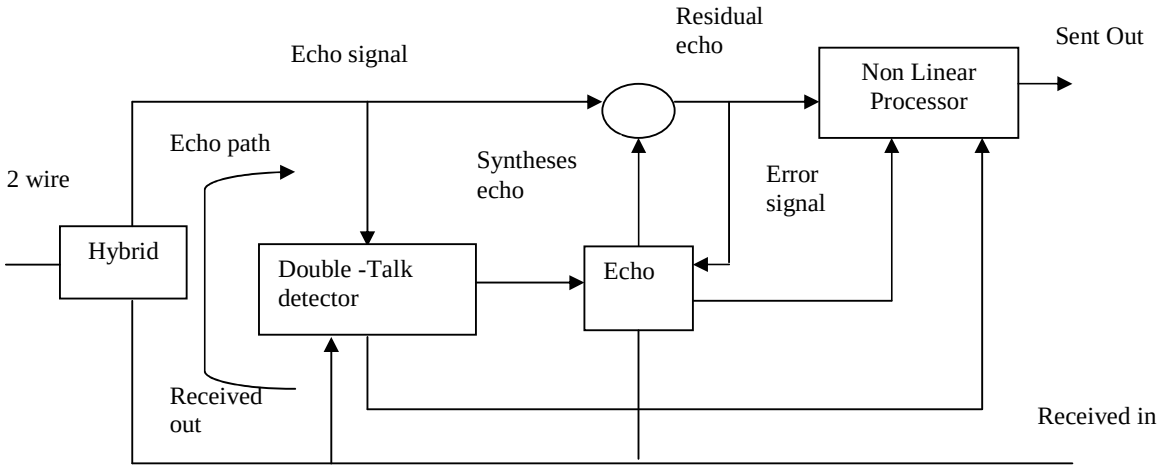


Fig: 1.7 Major Components of an Echo Canceller.

1.3.6.1 Echo Canceller Configuration

An echo canceller consists of a digital adaptive FIR filter arranged in a system identification configuration [7] as shown in fig 1.8. There are two important processes carried out within the echo canceller. The first process is the generation of the echo estimate by a convolution process between the input signal vector and the filter weight vector. The second process is the recursive update of the filter weight by an adaptive algorithm, where the NLMS algorithm is normally used. The echo estimate is generated as follows:

$$y(n) = \mathbf{X}^{\mathbf{T}}(n)\mathbf{W}(n) \qquad (1.1)$$

Consider that the echo path impulse response is time-invariant or at least slowly time varying , $\mathbf{H} = [h_0, h_{i-1}, \ldots, h_{i-N-1}]$. Again, by using the NLMS algorithm, the echo canceller filter weights can be updated as follows:

$$\mathbf{w}(n+1) = \mathbf{w}(n) + (\mu / \delta + \mathbf{x}^T(n)\mathbf{x}(n))\, e(n)\mathbf{x}(n) \qquad (1.2)$$

where μ is the step size and $\delta + \mathbf{x}^T(n)\mathbf{x}(n)$ is the normalization factor (a small positive constant δ has been appended to the denominator).

The error is defined as:

$$e(n) = \mathbf{x}^{\mathbf{T}}(n)(\mathbf{H} - \mathbf{W}(n)) + \eta(n) \qquad (1.3)$$

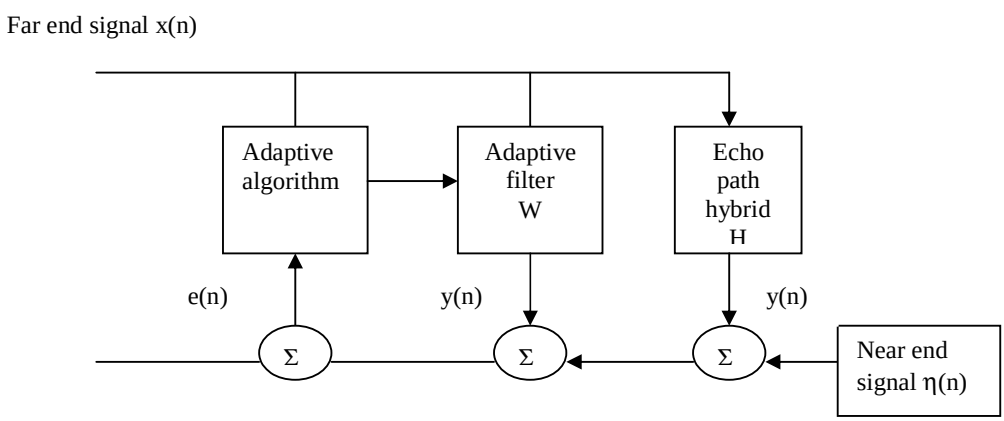


Fig1.8 Echo Canceller Configuration.

The choice of μ is critical for the good performance of the echo canceller. A large step size will in general provide a fast convergence speed and tracking ability but at the cost of a higher excess Mean Square Error. A small step size μ is needed for

doubletalk insensitivity and also to achieve small misadjustment in the steady state. Thus, this suggests a methodology on the automatic adjustment of μ, upon detecting the echo path change and double-talk [8]. Although this method demonstrates a fast convergence speed in computer simulation, it depends heavily on the efficient detection of the echo path change and double-talk. Under such conditions when fault detection happens, the echo cancellation will collapse. Thereby, a fixed step size is considered more robust while providing an adequate echo cancellation. By assigning a proportional gain to the tap weights magnitude the convergence speed can be quicker, hence producing a faster convergence for the echo cancellation. Nevertheless, these methods require more control mechanisms to detect the magnitude at each tap-weight and a limit of the individual tap loop gain is required to ensure stability. Moreover, the performance of the tap-weight adjustment algorithm is sensitive to the characteristic echo path impulse responses, which are required to track sudden changes. A combination of different orders of gradient descent-based algorithm is set to adapt in parallel to cover the possible different or time varying speech statistics.

1.4 Adaptive Filtering Algorithms

The gradient based adaptation starts with an old optimization technique known as the method of steepest descent. This has been discussed in the next chapter in detail. It is recursive in the sense that starting from some initial arbitrary value for tap weight vector, it improves with increasing number of iterations. The final value so computed for tap weight vector converges to Wiener solution. The LMS algorithm has been extensively analyzed in literature and a large number of results on its steady state misadjustment and tracking performance have been obtained, as appeared in [9,10]. The fixed step size least mean square (FSS LMS) algorithm is an important member of the family of stochastic gradient algorithms. The term ‘stochastic gradient’ is intended to distinguish it from the method of steepest descent that uses deterministic gradient in a recursive computation of the Wiener filter for stochastic inputs. This algorithm does not require measurements of the pertinent correlation functions, nor does it require matrix inversion. Subsequent works have discussed issue of optimization of step size or methods of varying step size to improve performance [11,12].

1.4.1 VSS-LMS Algorithm

As described in [13], the step size adjustment in this algorithm is controlled by Mean Square Error (MSE). The motivation is that a large MSE will cause step size to increase to provide faster tracking while a small MSE will result in a decrease in step size to yield smaller misadjustment. It is capable of giving both fast tracking as well as small misadjustment.

1.4.2 Modified VSS-LMS Algorithm

This algorithm is a modification of the already described Variable Step Size Least Mean Square (VSS-LMS) algorithm. It provides fast convergence at early stages of adaptation while ensuring small final misadjustment. The performance of the algorithm [14] is not affected by existing uncorrelated noise disturbances. Simulation results comparing this algorithm with the current variable step size algorithm clearly indicates its superior performance. For stationary environments, this algorithm performs better compared to VSS-LMS algorithm [13] and its performance is almost similar to the existing FSS LMS algorithm [9]. This algorithm has been used for echo cancellation and shows a better performance as compared to existing VSS-LMS algorithm.

1.4.3 Recursive Least Squares (RLS) Algorithm

The other class of adaptive filtering techniques is known as Recursive Least Squares (RLS) algorithms. These algorithms attempt to minimize the cost function

$$\phi(n) = \sum_{k=1}^{n} \lambda^{n-k} e_n^{\,2}(k) \tag{1.4}$$

Where k=1 is the time at which the RLS algorithm [7] commences and λ is a small positive constant very close to, but smaller than 1. With values of $\lambda<1$ 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. 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.

1.5 Performance of an Adaptive Algorithm

The factors that determine the performance of an algorithm are clearly stated below. Essentially, the most important factors as described in [15] are:

1. **Rate of Convergence:** This is defined as the number of iterations required for the algorithm to converge to its steady state mean square error. The steady state MSE is also known as the Mean asymptotic square error or MASE.
2. **Misadjustment:** This quantity describes steady-state behavior of the algorithm. This is a quantitative measure of the amount by which the ensemble averaged final value of the mean-squared error exceeds the minimum mean-squared error produced by the optimal Wiener filter. The smaller the misadjustment, the better the asymptotic performance of the algorithm.
3. **Numerical Robustness:** The implementation of adaptive filtering algorithms on a digital computer, which inevitably operates using finite word-lengths, results in quantization errors. These errors sometimes can cause numerical instability of the adaptation algorithm. An adaptive filtering algorithm is said to be numerically robust when its digital implementation using finite-word-length operations is stable.
4. **Computational Requirements**: This is an important parameter from a practical point of view. The parameters of interest include the number of operations required for one complete iteration of the algorithm and the amount of memory needed to store the required data and also the program. These quantities influence the price of the computer needed to implement the adaptive filter.
5. **Stability**: An algorithm is said to be stable if the mean-squared error converges to a final (finite) value.

Ideally, one would like to have a computationally simple and numerically robust adaptive filter with high rate of convergence and small misadjustment that can be implemented easily on a computer. In the applications of digital signal processing e.g. adaptive echo cancellation, the above factors play an important role.

1.6 Adaptive Echo Cancellation

An acoustic echo canceller can overcome the acoustic feedback that interferes with teleconferencing and hands free telecommunication. It adaptively identifies the transfer function between a loudspeaker and a microphone, and then produces an echo replica that is subtracted from the real echo [7]. Echo occurs when an audio source and sink operate in full duplex mode. In this situation the received signal is output through the telephone loudspeaker (audio source), this audio signal is then reverberated through the physical environment and picked up by the systems microphone (audio sink). The result is that time delayed and attenuated images of the original speech are returned to the distant user.

The present study deals with canceling these echo signals for improving the communication quality by using various adaptive filtering algorithms and comparing the performance of all these algorithms when applied to echo cancellation application. The simulations have been done in MATLAB.

Echo cancellation is critical to achieving high quality voice transmissions over packet networks, which typically face transmission delays above 30 to 40 ms. These long delays make echo readily apparent to listeners, and must be eliminated in order to provide viable telephony service [16].

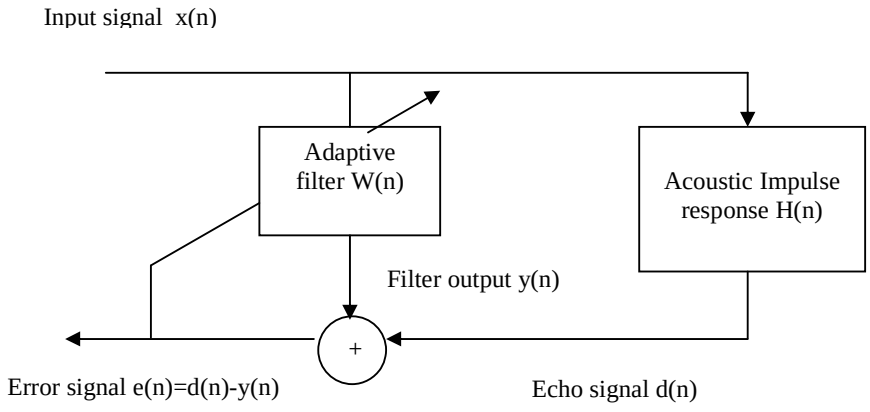


Fig 1.9: Block Diagram of an Adaptive Echo Cancellation System

The configuration of an acoustic Echo canceller is given in fig 1.9. The echo canceller identifies the transfer function of the acoustic echo path i.e. the impulse response H(n) between the loudspeaker and the microphone. Since the impulse response varies as a person moves and varies with the environment, an adaptive filter echo replica y(n) is then subtracted from the echo signal d(n) to give the error e(n)= d(n)-y(n).This is used to identify H(n). The desired signal is obtained by convolving the input signal with the impulse response of the acoustic environment and an echo replica is created at the output of the adaptive filter. The adaptive algorithm should provide real time operation, fast convergence, and high Echo Return Loss Enhancement (ERLE).

The various parameters that have been used for comparison of various adaptive algorithms in the present work when applied to echo cancellation application are:

1. Mean Square Error
2. Complexity
3. Convergence rate
4. Error estimated: The performance of the filter is determined by the size of the estimation error, that is, smaller the estimation error better the filter performance. As the estimation error approaches zero, the filter output approaches the desired signal.
5. Echo Return Loss Enhancement (ERLE): ERLE [5] is the ratio of send-in power and the power of a residual error signal immediately after the cancellation. It is measured in dB. ERLE measures the amount of loss introduced by the adaptive filter alone. ERLE depends on the size of the adaptive filter and the algorithm design. The higher the value of ERLE, the better the echo canceller. ERLE is a measure of the echo suppression achieved and is given by

$$ERLE = 10 \log_{10} (Pd/Pe)$$

6. Echo return loss (ERL): Measured in dB, ERL [2] is the ratio of receive-out and send- in power. ERL measures receive-out signal loss when it is reflected back as echo within the send-in signal.

$$ERL = 10 \log_{10}(Px/Pd)$$

1.7 Layout of Thesis

Chapter 2 deals with adaptive filter theory as it pertains to the application of adaptive filters.

Chapter 3 studies the basis of adaptive filtering techniques as well as the development and derivation of algorithms used within this thesis.

Chapter 4 discusses the modified variable step size algorithm and its advantages over the already existing variable step size algorithms in detail.

Chapter 5 describes the Echo canceller block diagram and configuration and discusses its use in hands free telephony and describes the simulations of adaptive filtering techniques as developed in MATLAB. It also shows the results of these simulations as well as discussing the advantages and disadvantages of each technique.

Chapter 6 summarizes the work and provides suggestions for future research

Appendices A and B: Appendix A discusses the Gaussian theorem in detail and Appendix B discusses the matrix inversion lemma used for the derivation of RLS algorithm.

Chapter 2

ADAPTIVE FILTER THEORY

2.1 Introduction

In the past three decades, interest in adaptive systems has increased, leading to widespread use of adaptive techniques in fields such as Communications, Signal Processing, Sonar and Biomedical Engineering. Adaptive systems adapt to the environment changes and search for the optimal system parameters based on a reference signal. In the case of a filter, the system parameters are the tap weights of the filter. The performance of an adaptive algorithm is highly dependent on the reference input and additive noise statistics. In the context of Wiener filter theory, there are assumptions of time invariance, linearity and Gaussian statistics such that the mean square error criteria will be the optimum cost function. These assumptions are often for the ease of mathematical analysis, but do not take into account of the broader problems of signals with non-Gaussian statistics. In the digital communication systems, efficient bandwidth utilization is economically important to maximizing profits, while at the same time maintaining performance and reliability. More importantly, the adaptive filter solution has to be relatively simple, which often leads to the use of the conventional Least Mean Square (LMS) algorithm. However, the performance of the LMS algorithm is often sub-optimal and the convergence rate is small. This, therefore, provides the motivation to explore and study variable step size LMS adaptive algorithms for various applications.

2.2 The Wiener Filter

These are a class of linear optimum discrete time filters known collectively as Wiener filters. Wiener filters are a special class of transversal Finite impulse response (FIR) filters that build upon the Mean Square Error (MSE) cost function to arrive at an optimal filter tap weight vector, which reduces the MSE signal to a minimum. Theory for a Wiener filter is formulated for general case of complex valued time series with filter specified in terms of its impulse response because baseband signal appears in complex form under many practical situations.

2.2.1 Mean Square Error Criterion

fig 2.1 illustrates a linear filter with the aim of estimating the desired signal d(n) from input x(n). Assume that d(n) and x(n) are samples of infinite length , random processes. In 'optimum filter design', signal and noise are viewed as stochastic processes. The filter is based on minimization of the mean square value of the difference between the actual filter output and some desired output, as shown in fig. 2.1:

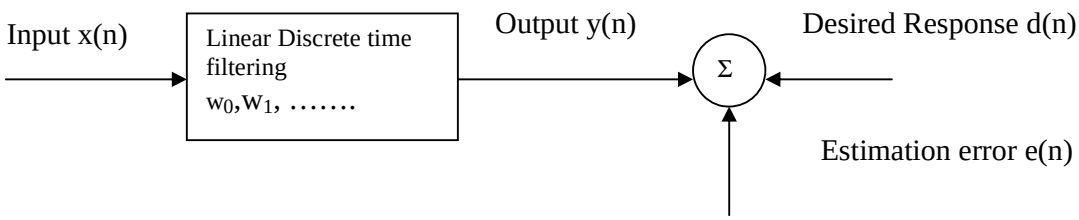


Fig: 2.1 Prototype Wiener Filtering Scheme

The requirement is to make the estimation error as small as possible in some Statistical sense by controlling impulse response coefficients $w_0, w_1, \ldots\ldots w_{N-1}$.

Two basic restrictions are:

1. The filter is linear, which makes mathematical analysis easy to handle.
2. The filter is an FIR (symmetrical and odd ordered) filter.

The filter output is y(n) and the estimation error is given by e(n). The performance of the filter is determined by the size of the estimation error, that is, a smaller estimation error indicates a better filter performance. As the estimation error approaches zero, the filter output y(n) approaches the desired signal d(n). Clearly, the estimation error is required to be as small as possible. In simple words, in the design of the filter parameters, an appropriate function of this estimation error as performance or cost function is chosen and the set of filter parameters is selected, which optimizes the cost function.

In Wiener filters, the cost function is chosen to be

$$\xi = E[e(n)^2] \qquad (2.1)$$

where E[.] denotes the expectation or ensemble average since both d(n) and x(n) are random processes.

2.3 Wiener Filter: Transversal, Real Valued Case

Consider an adaptive transversal filter as shown in Fig 2.2. Assume that the filter input x(n) and the desired response d(n) are real valued stationary processes. The filter tap weights w_0, w_1, ……w_{N-1} are also assumed to be real valued , where N equals the number of delay units or tap weights.

The filter input x(n) and tap weight vectors, w, can be defined as column vectors,

$$x(n) = [\ x(n) \quad x(n-1)\ldots\ldots x(n-N+1)]' \tag{2.2}$$

$$w = [w_0, w_1, \ldots\ldots\ldots\ldots\ldots w_{N-1}]' \tag{2.3}$$

The filter output is defined as

$$y(n) = \sum_{i=0}^{N-1} w_i x(n-i) = w'x(n) = x'(n)w \tag{2.4}$$

Subsequently, the error signal can be written as

$$e(n) = d(n)-y(n) = d(n) - w'\,x(n) = d(n) - x'(n)w \tag{2.5}$$

Substituting (2.5) into (2.1) , the cost function is obtained as,

$$\xi = E[(e(n)^2] = E[(d(n)- w'x(n))\,(d(n)-x'(n)w)] \tag{2.6}$$

Expanding the last expression of (2.6) we obtain,

$$\xi = E[\ d(n)^2] - E[\ d(n)x'(n)w] - E[d(n)\ w'x(n)] + E[w'\ x(n)\ x'\ (n)w] \tag{2.7}$$

Since w is not a random variable,

$$\xi = E[d(n)^2] - E[\ d(n)x'(n)]w - w'\ E[d(n)\ x(n)] + w'\ E[\ x(n)\ x'\ (n)]w \tag{2.8}$$

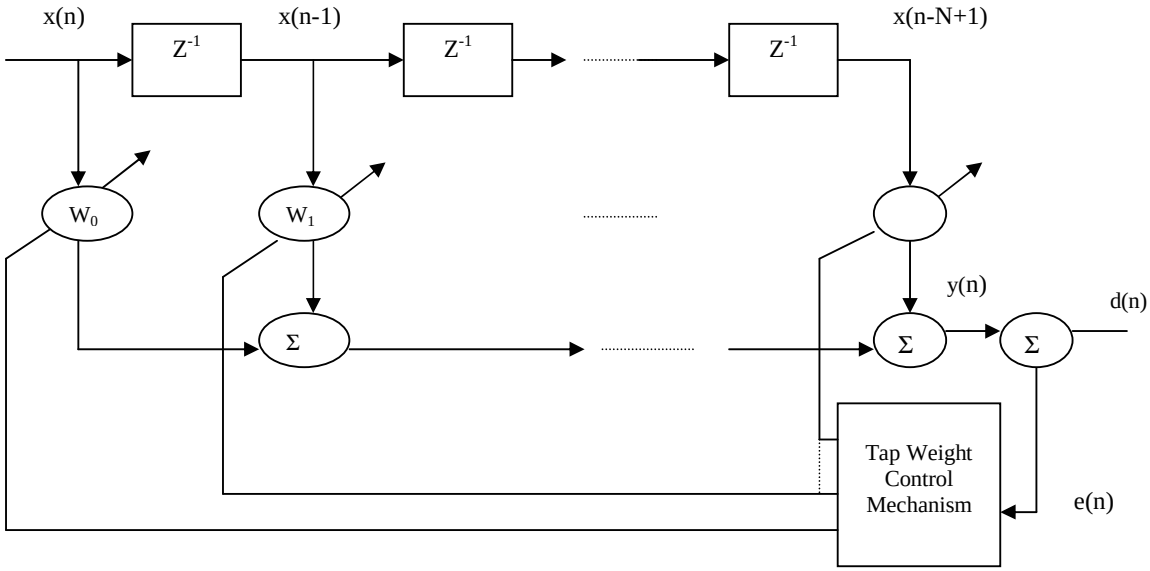


Fig2.2. Structure of an Adaptive Transversal Filter

Next, we can express E[d(n) x(n)] as an N x 1 cross correlation vector

$$p=E[d(n)x(n)]=[p_0, p_1, \ldots\ldots\ldots\ldots p_{N-1}]' \qquad (2.9)$$

And E[x(n) x′ (n)] as a N x N autocorrelation matrix R

$$R=E[x(n)x'(n)]=\begin{bmatrix} r_{00} & r_{01} & r_{02} & \ldots\ldots & r_{0,N-1} \\ r_{10} & r_{11} & r_{12} & \ldots\ldots & r_{1,N-1} \\ r_{20} & r_{21} & r_{22} & \ldots\ldots & r_{2,N-1} \\ . & . & . & . & . \\ r_{N-1,0} & r_{N-1,1} & r_{N-1,2} & \ldots\ldots & r_{N-1,N-1} \end{bmatrix} \qquad (2.10)$$

From (2.9), p′ = E[d(n) x′ (n)] and hence p′w = w′p

This implies that E[d(n) x′ (n)]w = E[d(n) x(n)]w′ . Subsequently, we get

$$\xi = E[d(n)^2] - E[d(n)x'(n)]w - w' E[d(n)x(n)] + w' E[x(n)x'(n)]w$$
$$= E[d(n)^2] - 2p'w + w' R w \qquad (2.11)$$

This is a quadratic function of tap weight vector 'w' with a single global minimum. To obtain the set of filter tap weights that minimizes the cost function, ξ, solve the system of equations that results from setting the partial derivatives of ξ with respect to every tap weight of the filter i.e. the gradient vector to zero. That is

$$\frac{\partial \xi}{\partial w_i} = 0 \tag{2.12}$$

For i = 0, 1, ……..N-1 where N = number of tap weights

The gradient vector in (2.12) can also be expressed as

$$\nabla \xi = 0 \tag{2.13}$$

where ∇ is the gradient operator defined as column vector

$$\nabla = \begin{bmatrix} \frac{\partial}{\partial w_o} & \frac{\partial}{\partial w_1} & \frac{\partial}{\partial w_{N-1}} \end{bmatrix} \tag{2.14}$$

and 0 on the right hand side of (2.13) denotes the column vector consisting of N zero. It has been further proved that the partial derivatives of ξ with respect to the filter tap weights can be solved such that

$$\nabla \xi = 2Rw - 2p \tag{2.15}$$

By letting ∇ξ = 0, the following equation is obtained, in which the optimum set of Wiener filter tap weights can be obtained,

$$Rw = p$$

This implies that

$$w = R^{-1} p = w_0 \tag{2.16}$$

Where w_o indicates the optimum tap weight vector. This equation is known as the Wiener Hopf equation and can be solved to obtain the tap weight vector, which corresponds to the minimum point of the cost function.

2.4 Iterative Search Algorithm

It has been shown in the previous section that the Wiener Hopf equation can be solved to obtain the optimum filter tap weights by minimizing a cost function, if the required statistics of the underlying signals 'R' and 'p' are available. Although this method is straightforward, it presents serious computational difficulties, especially when the filter contains a large number of tap weights and the input data rate is high.

An alternative is to use an iterative search algorithm [17] that starts at some arbitrary initial point in the tap weight vector space and moves progressively towards the optimum filter tap weight vector in steps. Each step is chosen with the aim of reducing the cost function. The principle of finding the optimum filter tap weight vector by progressive minimization of the underlying cost function by means of an iterative algorithm is central to the development of adaptive algorithms (e.g. LMS). In simplified terms, adaptive algorithms are actually iterative search algorithms derived for minimizing the cost function by replacing the true statistics with estimates obtained.

2.5 Method of Steepest Descent

Assume that the cost function to be minimized is convex (If the cost function corresponds to a convex quadratic surface, it has a unique minimum point. In other words, when the cost function is convex, the iterative search algorithm is guaranteed to converge to the optimum solution), we may start with an arbitrary point on the performance surface and take a small step in the direction in which the cost function decreases fastest. This corresponds to a step along the steepest descent slope of the performance at that point. Repeating this successively, convergence towards the bottom of the performance surface (corresponding to the set of parameters that minimize the cost function) is guaranteed.

The method of steepest descent [17] is an alternate iterative search method to find w_o (in contrast to solving the Wiener Hopf equation directly). The method of steepest

descent algorithm belongs to a family of iterative methods of optimization. It is a general scheme that performs an iterative search for a minimum point of any convex function of a set of parameters. Here, this method is implemented in transversal filter with the convex function referring to the cost function and the set of parameters referring to the filter tap weights. It uses the following procedures to search the minimum point of the cost function of a set of filter tap weights.

1. Begin with an initial guess of the filter tap weights whose optimum values are to be found for minimizing the cost function. Unless some prior knowledge is available, the search can be initiated by setting all the filter tap weights to zero, i.e. w(0).

2. Use this initial guess to compute the gradient vector of the cost function with respect to the tap weights at the present point.

3. Update the tap weights by taking step in the opposite direction (sign change) of the gradient vector obtained in step 2. This corresponds to step in the direction of the steepest descent in the cost function at the present input. Furthermore, the size of the step is chosen proportional to the size of the gradient vector.

4. Go back to Step 2, and iterate the process until no further significant change is observed in the tap weights i.e. the search has converged to an optimal point.

According to the above procedures, if w(n) is the tap weight vector at the nth iteration , then the following recursive equation may be used to update w(n).

$$w(n+1) = w(n) - \mu \nabla_n \xi \qquad (2.17)$$

where μ is the positive scalar called step size ,and $\nabla_n \xi$ denotes the gradient vector evaluated at the point w = w(n).

2.6 Error Performance Surface

The estimation error e(n) can be given as:

$$E(n)=d(n)-\sum_{i=0}^{N-1} w_i x(n-i) \tag{2.18}$$

The cost function can be written as:

$$\xi=E[d(n)^2]-\sum_{i=0}^{N-1} w_i^* E[x(n-i)d^*(n)]-\sum_{i=0}^{N-1} w_i E[x^*(n-i)d(n)] +$$

$$\sum_{i=0}^{N-1}\sum_{k=0}^{N-1} w_i^* w_k \; E[x^*(n-i)x^*(n-k)] \tag{2.19}$$

The cost function or the mean squared error is precisely a second order function of the tap weights in the filter. Since 'w' can assume a continuum of values in the N dimensional w-plane, the dependence of the cost function depends on the tap weights w_0, w_1, ……...w_{N-1} may be visualized as a bowl shaped (N+1)-dimensional surface with N degrees of freedom represented by the tap weights of the filter. The surface so described is called the error performance surface of the transversal filter. The surface is characterized by a unique minimum, where the cost function ξ attains its minimum value. At this point, the gradient vector $\nabla\xi$ is identically zero. The height ξ corresponds to the physical description of filtering the signal x(n-i) with the fixed filter weight w, from which a prediction error signal e(n) with power of ξ is generated. Some filter setting $w_o=(w_{o0},w_{o1})$ will produce the minimum MSE (w_0 is the optimum filter tap weight vector) . This theory is the base of basic adaptive algorithms of adaptive signal processing.

Chapter 3

ADAPTIVE FILTERING ALGORITHMS

3.1 Introduction

In the past three decades, interest in adaptive systems has increased, leading to widespread use of adaptive techniques in fields such as Communications, Signal Processing, Sonar and Biomedical Engineering. Adaptive systems adapt to the environment changes and search for the optimal system parameters based on a reference signal. In the case of a filter, the system parameters are the tap weights of the filter. The performance of an adaptive algorithm is highly dependent on the reference input and additive noise statistics. In the context of Wiener filter theory, there are assumptions of time invariance, linearity and Gaussian statistics such that the mean square error criteria will be the optimum cost function. These assumptions are often for the ease of mathematical analysis, but do not take into account of the broader problems of signals with non-Gaussian statistics. Alternatively, non-mean square criteria can be a better solution in non-Gaussian conditions .In the digital communication systems, efficient bandwidth utilization is economically important to maximize profits, while at the same time maintaining performance and reliability. More importantly, the adaptive filter solution has to be relatively simple, which often leads to the use of the conventional LMS (Least Mean Square) algorithm. However, the performance of the LMS algorithm is often sub-optimal if the aspect of non-Gaussianity is considered. This, therefore, provides the motivation to explore and study non-mean square adaptive algorithms for non-Gaussian conditions. Walach and Widrow studied the mean square and mean fourth to mean sixth error cost functions, which showed the possibility of using higher order algorithms in non-Gaussian conditions. The non-quadratic cost function and the least mean q-power error criteria have been used to study the convergence characteristics of different error metrics of cost function. The mix of mean square and mean fourth error cost function demonstrated the potential of adaptation in the presence of a mixture of different statistics. The above mentioned works have briefly summarized the current state of research of various orders of cost function. A chart categorizing the mean square and

the non-mean square algorithms is shown in fig 3.1. The classification of mean square and non-mean square algorithm is based on the gradient vector adaptation. A non-mean square algorithm is when the gradient vector is modified by mean of a function or mixed of other order gradient vector.

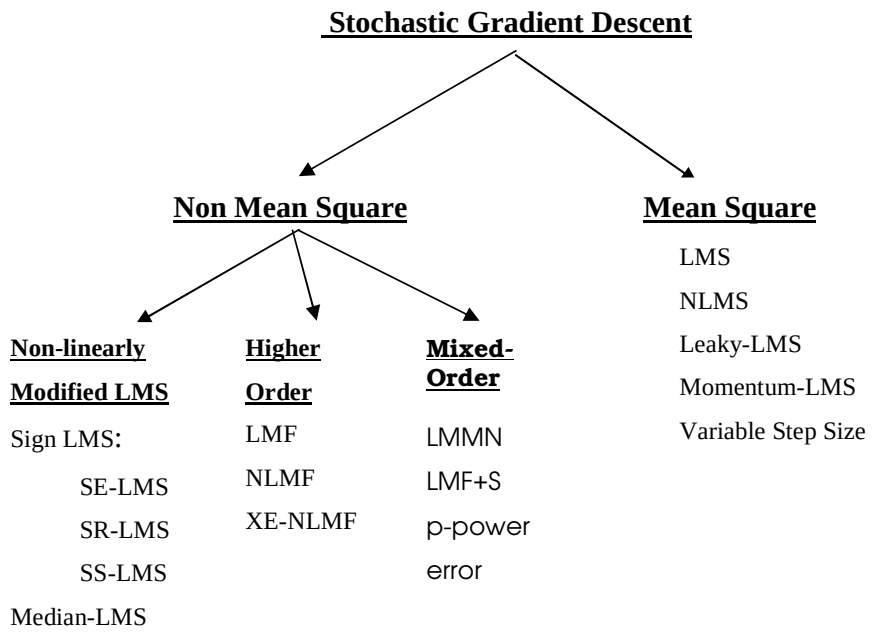


Fig3.1 The Grouping of LMS Offspring Algorithms

The performance of an adaptive filter is critically dependent not only on its internal structure, but also on the algorithm used to recursively update the filter weights that define the structure. Over the years, a number of stochastic-based non-mean square adaptive algorithms have been developed for different purposes, enriching the field of adaptive algorithms. Notably, the Least Mean Square algorithm is the most commonly used in practice, with non-mean square algorithms such as the Sign LMS, LMF and median-LMS, leakage and momentum-LMS algorithms also being important. The LMF (Least Mean Fourth) algorithm minimizes the mean fourth error cost function and can be regarded as a higher order algorithm that can outperform the LMS in excess mean square error, under a sub-Gaussian noise.

The gradient based adaptation starts with an odd optimization technique known as the 'Method of Steepest Descent' (As explained in section 2.5). In this, starting from some initial arbitrary value for the tap weight vector, it improves with increasing number of iterations. The final value so computed for tap weight vector converges to Wiener solution.

Another type of algorithms is the family of Stochastic Gradient algorithms. The term Stochastic gradient is intended to distinguish it from the method of steepest descent that uses deterministic gradient in a recursive computation of the Wiener filter for stochastic inputs. For any algorithm with a fixed value of step size for tap weight adaptation, there exists a tradeoff between the filter convergence rate and the steady state error. If a larger value of step size is used, then a faster convergence is attained, as long as the filter remains stable. On the other hand, the smaller the step size, the more accurate the estimation in the presence of observation noises.

Consequently, if we adaptively control the step size so that it stays large in the early stages of the filter convergence and becomes smaller as the convergence proceeds, both fast convergence and low estimation could be realized. Also the choice of step size reflects a tradeoff between misadjustment and speed of adaptation. From [18] a small step size gives a small misadjustment but a longer convergence time constant. A VSS-LMS algorithm, where step size adjustment is controlled by MSE, was proposed in [13]. Another robust modified variable step size LMS algorithm has been proposed in [14]. This algorithm provides fast convergence at early stages of adaptation while ensuring small final misadjustment. The performance of this algorithm is not affected by existing uncorrelated noise disturbances. The Median LMS algorithm proposed in [19] when the filter is subjected to input signals, which are corrupted by impulsive interference.

3.2 Adaptive Filtering Algorithms

3.2.1 LMS Algorithm

The LMS (Least Mean Square) [9] belongs to the class of gradient descent-based algorithms. It is the most widely used algorithm for numerous applications, especially channel equalization and echo cancellation. The simplicity and robustness of the LMS

updating equation are the most important features and leads to many successful developments of other gradient descent-based algorithms.

The LMS (least mean squares algorithm) is introduced as a way to recursively adjust the parameters of w(n) of a linear filter with the goal of minimizing the error between a given desired signal and the output of the linear filter. LMS is one of the many related algorithms which are appropriate for the task and whole family of algorithms have been developed which can address a variety of problem settings, computational restrictions and minimization criteria. This section begins with the derivation of the LMS algorithm as an instantaneous approximation to the steepest descent minimization of a cost function, which results in simple recursive scheme.

3.2.2 Derivation of LMS

The conventional adaptive LMS algorithm is a stochastic implementation of the method of steepest descent algorithm. Substitute the cost function into (2.17), we get

$$w(n+1)=w(n)-\mu\nabla_n e^2(n) \tag{3.1}$$

where ∇ is the gradient operator , $w = [w_0, w_1, \ldots\ldots\ldots\ldots w_{N-1}]'$ and μ is the step size .

using (2.14) and last term in (3.1) , we note that the ith element of $\nabla e^2(n)$ can be given by

$$\frac{\partial e^2(n)}{\partial w_i} = 2e(n)\frac{\partial e(n)}{\partial w_i} \tag{3.2}$$

The derivative of the first equality in (2.5) is

$$\frac{\partial e(n)}{\partial w_i} = 0 - \frac{\partial y(n)}{\partial w_i} \tag{3.3}$$

since d(n) is independent of w_i

Substituting (3.3) into (3.2)

$$\frac{\partial e^2(n)}{\partial w_i} = -2e(n)\frac{\partial y(n)}{\partial w_i} \tag{3.4}$$

Then take the derivative of y(n) in the first equality of (2.4) and substitute the result into (3.4)

$$\frac{\partial e^2(n)}{\partial w_i} = -2e(n)x(n-i) \tag{3.5}$$

Then using (2.14) and (3.5) , we get,

$$\nabla e^2(n)=-2e(n)x(n) \tag{3.6}$$

where x(n) = [x(n) x(n-1)……x(n-N+1)] ′ as stated in (2.2)

Substituting (3.6) into (3.1) gives

$$w(n+1)=w(n)+2\mu e(n)x(n) \tag{3.7}$$

This equation is known as the LMS tap weight adaptation. During every iteration, the tap weight adaptation updates the tap weight vector w in the direction of minimizing the cost function to find w_0.

It is noteworthy that the factor 2 in the last term of (3.7) is left out by some authors. If ½ is multiplied to both sides of (2.15) to cancel out the factor of 2 on the right hand side, we get ($\nabla\xi$ /2) = Rw – p. Substituting this into (2.17) leads to

$$w(n+1)=w(n)-\tfrac{1}{2}\mu\nabla_n\xi \tag{3.8}$$

Consequently,

$$w(n+1)=w(n)-\tfrac{1}{2}\mu\nabla e^2(n) \tag{3.9}$$

In summary, we can write the LMS algorithm for every search iteration, in the form of three operations:

1. Filter output : $y(n) = w\ x(n)$
2. Error Estimation : $e(n) = d(n) - y(n)$
3. Tap Weight adaptation: $w(n+1) = w(n) + \mu\ e(n)x(n)$

The second operation defines the estimation error e(n), computed based on the current estimate of the tap weight vector w in the first operation. The last term in the third operation refers to the correction that is applied to the previous estimate of the tap weight vector. (Corresponding to step 3 of the method of steepest descent).

The learning curve of LMS algorithm is given below. The learning curve of LMS is not smooth like the learning curve of steepest descent algorithm, as it has gradient noise due to statistical changes, as shown in fig 3.2

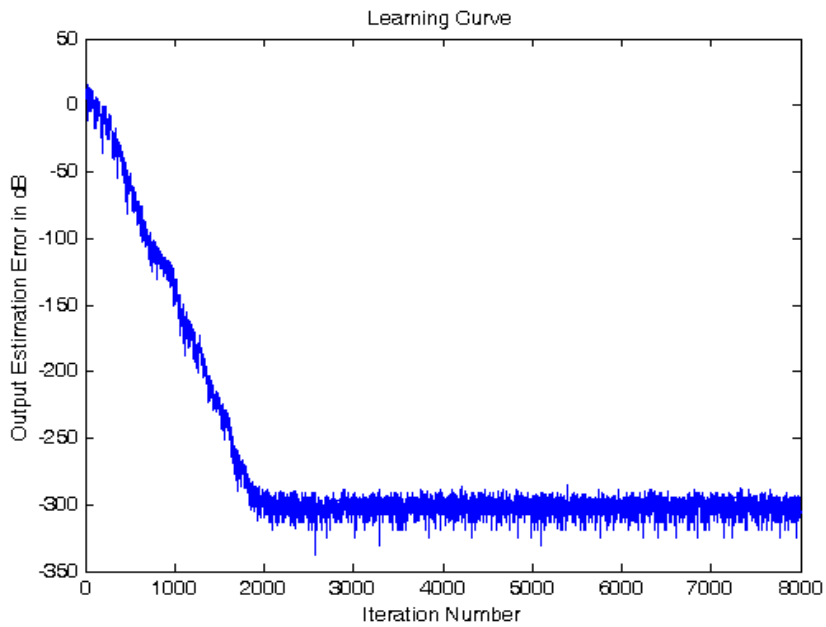


Fig 3.2 Learning curve of LMS algorithm

Some notable aspects of the performance of the Adaptive Filter are:

- LMS tends to reject the noisy data due to the smoothing action of the small step size parameter
- LMS can track slowly varying systems, and is often useful in non-stationary environments.

- The LMS error function has a unique global minimum, and hence the algorithm does not tend to get stuck at undesirable local minima.
- LMS is computationally simple (m multiplications and m additions per iteration) and memory efficient. (Only one m-vector must be stored).
- The convergence of LMS is often slow (it may take hundreds or thousands of iterations to converge from an arbitrary initialization).

3.2.3 Step Size Optimization

The step size plays an important role in controlling the performance of the LMS adaptive filter. The problem of the conventional LMS algorithm is that the fixed step size governs the trade-off between the convergence rate and the steady state error. A large step reduces the transient time but will result in a larger steady state mean square error. On the other hand, to achieve a smaller steady state error, a small step size has to be used which will cause a slower convergence rate. Selection of the step size is application specific with priority requirements such as fast convergence, robustness, tracking capability and adaptation accuracy. To achieve overall high performance, a sequence of optimum step sizes can be used to imitate the RLS performance; known as the time-varying step size algorithm. It is possible that the optimum step size can be determined by solving the difference equation for the minimum MSE at each iteration. A number of time varying step sizes have been proposed to improve the step size trade-off effect, which are a vital tool in achieving desired adaptive performance.
To ensure stability (or convergence) of the LMS algorithm, the step size parameter is bounded by the following equation [2]:

$$0 < \mu < (2 / \text{tap weight power}) \tag{3.10}$$

where tap weight input power is the sum of the mean squared values of all the tap inputs in the transversal filter and is given by $\sum_{k=0}^{N-1} E\left[|x(n-k)|^2\right]$. Note that the upperbound is dependent on the statistics of filter input signals. Intuitively, we may interpret from this equation that when the power of the input signals varies greatly, a smaller step size is required to avoid instability or gradient noise amplification.

3.2.4 Effect of Power Spectral Density of the Input Signal

The convergence rate of the LMS algorithm deteriorates with higher input correlation levels due to a greater interaction among the adaptive tap coefficients. To ensure improved convergence rate, the LMS algorithm requires input signals to have equal excitation over the whole range of frequency.

3.2.5 Effect of Filter Tap Length

In order to ensure good asymptotic performance, the length of the LMS adaptive FIR filter must be sufficient to cover the impulse response of the unknown channel [18]. However, this may lead to increased computational complexity when the impulse response of the unknown channel is ‘ long’. Moreover, this may lead to poor convergence rates when the input signals are highly correlated [18].

3.2.6 Effect of Input Signal Power

The correction term $\mu e(n)x(n)$ in the LMS tap weight adaptation is directly proportional to the tap input x(n). In other words, the convergence rate and stability of the LMS algorithm is directly dependent on the value of $\mu\sigma_x^{\ 2}$, where $\sigma_x^{\ 2}$ is the variance of power of the input signal [18].

When the power of the input signal x(n) is large or varies greatly, the LMS algorithm demonstrates an unstable behavior known as gradient noise amplification. Consequently, the LMS algorithm becomes unstable and therefore will not lead to the optimal solution. To deal with this problem, a modified version of LMS known as Normalized algorithm can be implemented.

3.2.7 Normalized LMS (NLMS) Algorithm

Determining the upper bound step size is a problem for the variable step size algorithm if the input signal to the adaptive filter is non-stationary. The fastest convergence is achieved with the choice of step size as follows:

$$\mu_{max} = 1 / (\lambda_{max} + \lambda_{min}) \qquad (3.13)$$

However, experimental results have shown that the maximum step size in equation (3.13) does not always produce the stable and fast convergence, according to Kwong

[13], (2/3)μ_{max} *is* a rule of thumb for LMS algorithm. To increase the convergence speed Normalized LMS (NLMS) algorithm is a natural choice [7], because the step size is normalized by the input signal power. The NLMS is always the favorable choice of algorithm for fast convergence speed and for non-stationary input. The value of $\mu\sigma_x^2$ directly affects the convergence rate and stability of the LMS filter. As the name implies, the NMLS algorithm is an effective approach to overcome this dependence, particularly when the variation of the input signal power is large, by normalizing the update step size with an estimate of the input signal variance. In practice, the correction term applied to the estimated tap weight vector w(n) at the n-th iteration is normalized with respect to squared Euclidean norm of the tap input x(n) at the (n-1)th iteration,

$$w(n+1) = w(n) + \frac{\alpha}{\|x(n)\|^2} e(n)x(n) \tag{3.14}$$

where $\| \ \|^2$ = Euclidean Norm

Apparently, the convergence rate of the NMLS algorithm is directly proportional to the NLMS adaptation constant α, i.e. the NLMS algorithm is independent of the input signal power. By choosing α so as to optimize the convergence rates of the algorithms, the NLMS algorithm converges more quickly than the LMS algorithm.

It can also be stated that the NLMS is convergent in mean square if the adaptation constant α is from 0 to 2 (however a more practical step size for NLMS is always less than unity).

$$0 < \alpha < 2$$

Despite this particular edge that the NLMS exhibits, it does have a slight problem of its own. When the input vector x(n) is small, instability may occur since we are trying to perform numerical division by small value of the Euclidean Norm

However, this can be easily overcome by appending a positive constant to the denominator in (3.10) such that

$$w(n+1)=w(n) + \frac{\alpha}{c+\|x(n)\|^2} e(n)x(n) \tag{3.15}$$

Where c+$\|x(n)\|^2$ is the normalization factor. With this, more robust and reliable implementation of the NLMS algorithm is obtained.

3.2.8 RLS Algorithm

The major advantage of the LMS algorithm lies in its low computational complexity. However, the price paid for this simplicity is slow convergence, especially when the eigen values of the auto correlation matrix have a large spread, that is when $\lambda_{max}/\lambda min$ >>1. From another point of view, the LMS algorithm has only a single adjustable parameter for controlling the convergence rate, namely, the step size parameter. Since this is limited for purposes of stability to be less that the upperbound, the modes corresponding to the eigen values converge slowly. To obtain faster convergence, it is necessary to devise more complex algorithms, which involve additional parameters. In deriving more rapidly converging adaptive filtering algorithms, the least squares criterion instead of the statistical approach based on the MSE criterion is adopted.

The cost function for RLS algorithm is given as:

$$\phi(n) = \sum_{k=1}^{n} \lambda^{n-k} e_n^{\ 2}(k) \tag{3.16}$$

The RLS cost function of equation 3.16 shows that at a time n, all previous values of the estimation error since the commencement of the RLS algorithm are required. Clearly as time progresses the amount of data required to process this algorithm increases. The fact that memory and computation capabilities are limited makes the RLS algorithm [20] a practical impossibility in its purest form. However, the derivation still assumes that all data values are processed. In practice only a finite number of previous values are considered, this number corresponds to the order of the RLS FIR filter, N.

In equation 3.16, 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 [21].

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.

3.2.9 Derivation of RLS Algorithm

In the RLS algorithm, we deal with the data sequence X(n) directly and obtain estimates of correlations from the data. Since the algorithms will be recursive in time, it is also necessary to introduce a time index in the filter coefficient vector and in the error sequence. First we define $y_n(k)$ as the output of the FIR filter, at n, using the current tap weight vector, and the input vector of a previous time k. The estimation error value $e_n(k)$ is the difference between the desired output value at time k, and the corresponding value of $y_n(k)$. The following equations give the expression for the error and the desired signal.
for k=1, 2, 3,., n.

$$\begin{aligned} y_n(k) &= w^T(n)x(k) \\ e_n(k) &= d(k) - y_n(k) \\ e(n) &= d(n) - y(n) \end{aligned} \tag{3.17}$$

X(n) can be defined as the matrix consisting of the n previous input column vector up to the present time then **y**(n) can also be expressed as equation 3.18

$$\begin{aligned} X(n) &= [x(1), x(2), \ldots\ldots\ldots x(n)] \\ y(n) &= X^T(n)w(n) \\ d(n) &= [d(1), d(2) \ldots\ldots\ldots d(n)]^T \\ y(n) &= [y_n(1), y_n(2) \ldots\ldots y_n(n)]^T \\ e(n) &= [e_n(1), e_n(2) \ldots\ldots e_n(n)]^T \end{aligned} \tag{3.18}$$

The cost function of equation 3.16, can then be expressed in matrix vector form using $\Lambda(n)$, a diagonal matrix consisting of the weighting factors.

where

$$\phi(n) = \sum_{k=1}^{n} \lambda^{n-k} e_n^{\,2}(k) \tag{3.19}$$

Substituting values from equations 3.17 and 3.1 cost function can be expanded then reduced as in equation 3.20. (Temporarily dropping (n) notation for clarity).

$$\begin{aligned}
\phi(n) &= \sum_{k=1}^{n} \lambda^{n-k} e_n^{\,2}(k) \\
&= e^T(n)\Lambda'(n)e(n) \\
&= d^T\Lambda' d - d^T\Lambda' y - y^T\Lambda' d + y^T\Lambda' y \\
&= d^T\Lambda' d - d^T\Lambda'(X^T w) - (X^T w)^T\Lambda' d + (X^T w)^T\Lambda'(X^T w) \\
&= d^T\Lambda' d - 2\alpha_\lambda'^{\,T} w + w^T \beta_\lambda' w
\end{aligned} \tag{3.20}$$

where

$$\begin{aligned}
\alpha_\lambda'(n) &= X(n)\Lambda'(n)d(n) \\
\beta_\lambda'(n) &= X(n)\Lambda'(n)X^T(n) \\
w'(n) &= \beta_\lambda'^{-1}(n)\alpha_T'(n)
\end{aligned} \tag{3.21}$$

Then derive the gradient of the above expression for the cost function with respect to the filter tap weights. By forcing this to zero we then find the coefficients for the filter, $w'(n)$ which minimizes the cost function.

$$\begin{aligned}
\alpha_\lambda'(n) &= \lambda\alpha_\lambda'(n-1) + x(n)d(n) \\
w'(n) &= \beta_\lambda'^{-1}(n)\alpha_\lambda'(n)
\end{aligned} \tag{3.22}$$

The matrix $\beta(n)$ in the above equation can be expanded and then rearranged in a recursive form, then the special form of the matrix inversion lemma can be used (Appendix B) to find an inverse for this matrix, which is required to calculate the tap weight vector update. The vector

k(n) is known as the gain vector and is included in

$$\beta_{\lambda}'^{-1}(n) = \lambda\beta_{\lambda}'^{-1}(n-1) + x(n)x^{T}(n)$$

$$= \lambda^{-1}\beta_{\lambda}'^{-1}(n-1) - \frac{\lambda^{-2}\beta_{\lambda}'^{-1}(n-1)x(n)x^{T}(n)\beta_{\lambda}'^{-1}(n-1)}{1+\lambda^{-1}x^{T}(n)\beta_{\lambda}'^{-1}(n-1)x(n)}$$

$$= \lambda^{-1}(\beta_{\lambda}'^{-1}(n-1) - k(n)x^{T}(n)\lambda\beta_{\lambda}'^{-1}(n-1))$$

order to simplify the calculation.

(3.23)

Where

$$k(n) = \frac{\lambda^{-1}\beta_{\lambda}'^{-1}(n-1)x(n)}{1+\lambda^{-1}x^{T}(n)\beta_{\lambda}'^{-1}(n-1)x(n)} \tag{3.24}$$

$$= \beta_{\lambda}'^{-1}(n)x(n)$$

The vector $\alpha'_{\lambda}(n)$ of equation 3.22 can also be expressed in a recursive form. Using this and substituting $\beta^{-1}(n)$ from equation 3.23 into equation 3.22 we can finally arrive at the filter weight update vector for the RLS algorithm, as in equation 3.25.

$$= \beta_{\lambda}'^{-1}(n-1)\alpha_{\lambda}'(n-1) - k(n)x^{T}\beta_{\lambda}'^{-1}(n-1)\alpha_{\lambda}'(n-1) + k(n)d(n)$$
$$= w'(n-1) - k(n)x^{T}(n)w'(n-1) + k(n)d(n)$$
$$= w'(n-1) + k(n)(d(n) - w'(n-1)x(n))$$

where

$$e_{n-1}'(n) = d(n) - w'^{T}(n-1)x(n)$$

(3.25)

Each iteration of the RLS algorithm requires $4N^2$ multiplication operations and $3N^2$

$$w'(n) = w'(n-1) + k(n)e_{n-1}'(n)$$

(3.26)

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.

Chapter 4

MODIFIED VARIABLE STEP SIZE LMS ALGORITHM

4.1 Introduction

There is a vast amount of literature on variable step size methods, which exploit the trade-off between fast convergence and lower steady state error. Gear shifting is a popular approach by Widrow, which is based on using large step size values when the filter weights are far from the optimal solution and small step size values when near to the optimum solution. A number of time-varying step-size algorithms have been proposed to enhance the performance of the conventional LMS algorithm Several criteria have been used: squared instantaneous error, sign changes of successive samples of the gradient, attempting to reduce the squared error at each instant, or cross correlation of input and error. The variable step size algorithm given by Kwong and Johnston has been discussed in the next section and its modification has also been proposed. The modified VSS LMS algorithm gives a better performance as compared to other variable step size algorithms.

4.2 Variable Step Size LMS Algorithm

Based on the error-squared power, Kwong and Johnston proposed a simpler Variable Step Size least mean square algorithm (VSS LMS)[13]. The error power reflects the convergence state of the adaptive filter, where a converging system has a higher error power while the converged system has a smaller error power. Therefore, scalar step

size increases or decreases as the squared error increases or decreases, thereby allowing the adaptive filter to track changes in the system and produces a smaller steady state error. The step size of the VSS algorithm is adjusted as follows:

$$\mu(n+1)=\alpha\mu(n)+\beta\, e^2(n) \tag{4.1}$$

The variable step size algorithm, as appeared in [13] is of the form

$$w(n+1)=w(n)+\mu(n)x(n)e(n) \tag{4.2}$$

where $e(n)=d(n)-x(n)^T w(n)$ (4.3)

Step size is updated as

$$\mu'(n+1) = \alpha\mu(n)+\beta e^2(n) \quad \text{with } 0< \alpha<1 ,\ \beta >0 \tag{4.4}$$

and $\mu(n+1)= \mu_{max}$ if $\mu'(n+1)> \mu_{max}$

μ_{min} if $\mu'(n+1)< \mu_{min}$

$\mu'(n+1)$ otherwise

where $0<\mu_{min}<\mu_{max}$

To ensure stability, the variable step size $\mu(n)$ is constrained to the pre-determined maximum and minimum step size values of the LMS algorithm, while α and β are the parameters controlling the recursion. Simply, the VSS algorithm's step size value change by tracking the error square or the error power. A large error increases the step size to provide faster tracking while a small error reduces the step size for smaller steady state error. Although this approach can improve the step size trade-off effect, the drawback is that the maximum and minimum step sizes would require to be known a priori. This is essential in order to achieve the fastest convergence rate while not causing instability.

A different technique usually known as the gradient adaptive step size, is as follows [22]:

$$\mu(n)=\mu\,(n-1) + \phi\, e(n)e(n-1)\mathbf{X}^{\mathbf{T}}(n-1)\mathbf{X}(n) \tag{4.5}$$

where ϕ is a small positive constant which controls the recursion. The recursion of $\mu(n)$ in equation 4.5 is such that the gradient $(e(n)\mathbf{X}^{\mathbf{T}}(n))$ is larger during the converging period while becoming zero after convergence.

The variable step size algorithms (except for the gradient adaptive step size) are based on some heuristic rules on the step size adjustment which are translated into numerical formulae. An overall weakness of the variable step size algorithms are that they require the user to select additional step size recursion constants and an initial step size to control the adaptive behavior of the step size sequence. More importantly, anticipation of the maximum and minimum limits are needed to avoid instability as well as to maximize performance. Nevertheless, the body of work in this field has enabled adaptive performance to be achieved that is comparable to the RLS algorithm. However, the performances gained are always followed by an increase in complexity and additional parameters to manage. The gear shifting approach and the VSS algorithm are the most simple and effective algorithms for fast convergence. Alternatively, to increase the convergence speed, the normalized LMS algorithm is the natural choice, as it is independent of the parameter selection.

4.3 Modified Variable Step Size LMS Algorithm

A number of time-varying step-size algorithms have been proposed to enhance the performance of the conventional LMS algorithm. Experimentation with these algorithms indicates that their performance is highly sensitive to the noise disturbance [23]. The present work discusses a robust variable step-size LMS-type algorithm providing fast convergence at early stages of adaptation while ensuring small final misadjustment. The performance of the algorithm is not affected by existing uncorrelated noise disturbances. Simulation results comparing the proposed algorithm to current variable step-size algorithm clearly indicate its superior performance.

Since its introduction, the LMS algorithm has been the focus of much study due to its simplicity and robustness, leading to its implementation in many applications. It is well known that the final excess Mean Square Error (MSE) is directly proportional to the adaptation step size of the LMS while the convergence time increases as the step size decreases. This inherent limitation of the LMS necessitates a compromise between the opposing fundamental requirements of fast convergence rate and small

misadjustment demanded in most adaptive filtering applications. As a result, researchers have constantly looked for alternative means to improve its performance. One popular approach is to employ a time varying step size in the standard LMS weight update recursion. This is based on using large step-size values when the algorithm is far from the optimal solution, thus speeding up the convergence rate. When the algorithm is near the optimum, small step-size values are used to achieve a low level of misadjustment, thus achieving better overall performance. This can be obtained by adjusting the step-size value in accordance with some criterion that can provide an approximate measure of the adaptation Process State.

Several criteria have been used:

- Squared instantaneous error [13]
- Sign changes of successive samples of the gradient [24]
- Attempting to reduce the squared error at each instant [25]
- Cross correlation of input and error [26].

Experimental results show that the performance of existing variable step size (VSS) algorithms is quite sensitive to the noise disturbance. Their advantageous performance over the LMS algorithm is generally attained only in a high signal-to-noise environment. This is intuitively obvious by noting that the criteria controlling the step-size update of these algorithms are directly obtained from the instantaneous error that is contaminated by the disturbance noise. Since measurement noise is a reality in any practical system, the usefulness of any adaptive algorithm is judged by its performance in the presence of this noise. The performance of the VSS algorithm deteriorates in the presence of measurement noise. Hence a new VSS LMS algorithm is proposed, where the step size of the algorithm is adjusted according to the square of the time-averaged estimate of the autocorrelation of e(n) and e(n-1) . As a result, the algorithm can effectively adjust the step size as in [13] while maintaining the immunity against independent noise disturbance. The MVSS LMS algorithm allows more flexible control of misadjustment and convergence time without the need to compromise one for the other.

4.3.1 Algorithm Formulation

In [13], the adaptation step size is adjusted using the energy of the instantaneous error. The weight update recursion is given by

$$W(n+1) = W(n) + \mu(n)e(n)X(n) \tag{4.6}$$

and the step-size update expression is

$$\mu(n+1) = \alpha\mu(n) + \gamma e^2(n) \tag{4.7}$$

where , $0< \alpha<1$, $\gamma>0$ and $\mu(n+1)$ is set to μ_{min} or μ_{max} when it falls below or above these lower and upper bounds, respectively. The constant μ_{max} is normally selected near the point of instability of the conventional LMS to provide the maximum possible convergence speed. The value of μ_{min} is chosen as a compromise between the desired level of steady state misadjustment and the required tracking capabilities of the algorithm. The parameter γ controls the convergence time as well as the level of misadjustment of the algorithm. The algorithm has preferable performance over the fixed step-size LMS: At early stages of adaptation, the error is large, causing the step size to increase, thus providing faster convergence speed. When the error decreases, the step size decreases, thus yielding smaller misadjustment near the optimum. However, using the instantaneous error energy as a measure to sense the state of the adaptation process does not perform as well as expected in the presence of measurement noise. This can be seen from (4.7). The output error of the identification system is

$$e(n) = d(n) - X^T(n)W(n) \tag{4.8}$$

where the desired signal is given by

$$d(n) = X^T(n)W^*(n) + \xi(n) \tag{4.9}$$

$\xi(n)$ is a zero-mean independent disturbance, and W*(n) is the time-varying optimal weight vector. Substituting (4.8) and (4.9) in the step-size recursion, we get

$$\mu(n+1) = \alpha\mu(n) + \gamma V^T(n)X(n)X^T(n)V(n) + \gamma\xi^2(n) - 2\gamma\xi(n)V^T(n)X(n) \tag{4.10}$$

where V(n) = W(n)-W*(n) is the weight error vector. The input signal autocorrelation

matrix, which is defined as $R = E\{X(n)X^T(n)\}$, can be expressed as $R = Q\Lambda Q^T$, where Λ is the matrix of eigenvalues , and Q is the modal matrix of R. Using $V(n) = Q^T V(n)$ and $X(n) = Q^T X(n)$, then the statistical behavior of $\mu(n+1)$ is determined by taking the expected average of (4.10)

$$E\{\mu(n+1)\} = \alpha E\{\mu(n)\} + \gamma(E\{\xi^2(n)\} + E\{V^T(n)\Lambda V(n)\}) \tag{4.11}$$

where the common independence assumption of V(n) and X(n) has been used . Clearly, the term $E\{V^T(n)\ \Lambda V(n)\}$ influences the proximity of the adaptive system to the optimal solution, and $\mu(n+1)$ is adjusted accordingly. However, due to the presence of $E\{\xi^2(n)\}$, the step-size update is not an accurate reflection of the state of adaptation before or after convergence. This reduces the efficiency of the algorithm significantly. More specifically, close to the optimum, $\mu(n)$ will still be large due to the presence of the noise term $E\{\xi^2(n)\}$. This results in large misadjustment due to the large fluctuations around the optimum. Therefore, a different approach is proposed to control step-size adaptation. The objective is to ensure large $\mu(n)$ when the algorithm is far from the optimum with $\mu(n)$ decreasing as we approach the optimum even in the presence of this noise. The proposed algorithm achieves this objective by using an estimate of the autocorrelation between e(n) and e(n-1) to control step-size update. The estimate is a time average of e(n)e(n-1) that is described as

$$p(n) = \beta p(n-1) + (1-\beta)e(n)e(n-1) \tag{4.12}$$

The use of p(n) in the update of $\mu(n)$ serves two objectives. First, the error autocorrelation is generally a good measure of the proximity to the optimum. Second, it rejects the effect of the uncorrelated noise sequence on the step-size update. In the early stages of adaptation, the error autocorrelation estimate $p^2(n)$ is large, resulting in a large $\mu(n)$. As we approach the optimum, the error autocorrelation approaches zero, resulting in a smaller step size. This provides the fast convergence due to large initial $\mu(n)$ while ensuring low misadjustment near optimum due to the small final $\mu(n)$ even in the presence of $\xi(n)$. Thus, the proposed step size update is given by

$$\mu(n+1) = \alpha\mu(n) + \gamma p(n)^2 \tag{4.13}$$

where limits on μ(n+1), α and γ are the same as those of the VSS LMS algorithm in [13]. The positive constant β (0< β <1) is an exponential weighting parameter that governs the averaging time constant, i.e., the quality of the estimation. In stationary environments, previous samples contain information that is relevant to determining an accurate measure of adaptation state, i.e., the proximity of the adaptive filter coefficients to the optimal ones. Therefore, β should be approximately equal to 1. For nonstationary optimal coefficients, the time averaging window should be small enough to allow for forgetting of the deep past and adapting to the current statistics, i.e., β<1.

The step size in (4.13) can be rewritten as

$$\mu(n+1) = \alpha\mu(n) + \gamma[E\{V^T(n)X(n)X^T(n-1)V(n-1)\}]^2 \tag{4.14}$$

Assuming perfect estimation of the autocorrelation of e(n) and e(n-1), we note that as a result of the averaging operation, the instantaneous behavior of the step size will be smoother. It is also clear from (4.14) that the update of μ(n) is dependent on how far we are from the optimum and is not affected by independent disturbance noise. Finally, the proposed algorithm involves two additional update equations ((4.12) and (4.13)) compared with the standard LMS algorithm. Therefore, the added complexity is six multiplications per iteration. Compared with the VSS LMS algorithm in [13], the algorithm adds a new equation (4.12) and a corresponding parameter β.

4.3.2 Analysis of Modified VSS LMS Algorithm

The performance analysis of the modified VSS LMS algorithm is given in this section. The input signal is assumed to be a zeromean, stationary Gaussian. The analysis is conducted assuming exact modeling of the unknown system, i.e., the number of coefficients of the optimal weight vector W*(n) is the same as the number of the coefficients of the adaptive filter W(n) . In addition, the measurement noise sequence, which is represented by ξ(n) in (4.9), is zero-mean and white. The weight

update recursion of the proposed algorithm is given in (4.6), where e(n) is defined in (4.8) and (4.9), and μ(n) is defined in (4.12) and (4.13). The nonstationary environment is modeled by a time-varying optimal weight vector generated by a random walk model [2] as

$$W(n) = W(n-1) + \eta(n-1) \tag{4.15}$$

where η(n) is a stationary noise process of zero mean and correlation matrix $\sigma_n^2 I$. For a stationary system , $\sigma_n^2 = 0$, and W*(n)=W* . Substituting (4.8), (4.9), and (4.15) in (4.6) results in (4.16)

$$V'(n+1) = [I - \mu(n)X'(n)X'^T(n)]V'(n) + \mu(n)\xi(n)X'(n) - \eta'(n) \tag{4.16}$$

where $\eta'(n) = Q^T\eta(n)$. Normally, γ is chosen to be a very small value; hence, μ(n) is slowly varying when compared with e(n) and X(n) . This justifies the independence assumption of μ(n) and $\mu^2(n)$ with e(n), W(n) , and X(n). Accordingly, the following condition can be obtained from (4.16) to ensure convergence of the weight vector mean

$$0 < E\{\mu(n)\} < \frac{2}{\lambda_{\max}} \tag{4.17}$$

where λ_{max} is the maximum eigenvalue of R . However, convergence of the mean weight vector cannot guarantee convergence of the mean square error. Therefore, conditions that will ensure convergence in the mean square sense need to be determined. To evaluate the performance of the system, approximate expressions for misadjustment are derived. This leads to tight conditions for the selection of the parameters α, β, and γ. Finally, some guidelines are selected to select γ to guarantee MSE convergence while producing the desirable misadjustment level. The MSE is given by [9]

$$\begin{aligned} E\{e^2(n)\} &= \varepsilon_{\min} + \varepsilon_{ex}(n) \\ &= \varepsilon_{\min} + E\{V'^T(n)\Lambda V'(n)\} \end{aligned} \tag{4.18}$$

where $\varepsilon_{ex}(n)$ is the excess MSE, and $\varepsilon_{min} = E\{\xi^2(n)\}$ is the minimum value of the MSE. Equation (4.18) shows that the MSE is directly related to the diagonal elements of $E\{V'(n)V'^T(n)\}$. Consequently, stability of the MSE is ensured by the stability of these elements. Postmultiplying both sides of (4.16) by $V'^T(n+1)$ and then taking the expected value yields

$$\begin{aligned}
&E\{V'(n+1)V'^T(n+1)\} = E\{V'(n)V'^T(n)\} - E\{\mu(n)\}E\{V'(n)V'^T(n)\}\Lambda \\
&- E\{\mu(n)\}\Lambda E\{V'(n)V'^T(n)\} + 2E\{\mu^2(n)\}\Lambda E\{V'(n)V'^T\}\Lambda \\
&+ E\{\mu^2(n)\}\Lambda tr(\Lambda E\{V'(n)V'^T(n)\}) + E\{\mu^2(n)\}\varepsilon_{\min}\Lambda + \sigma^2{}_n I
\end{aligned} \tag{4.19}$$

Note that in (4.19), the Gaussian factoring theorem (Appendix A) has been used to simplify the expression $E\{X'(n)X'^T(n)V'(n)V'^T(n)X'(n)X'^T(n)\}$ into a sum of second-order moments . From (4.12), p(n) can be obtained recursively as

$$p(n) = (1-\beta)\sum_{i=0}^{n-1} \beta^i e(n-i)e(n-i-1) \tag{4.20}$$

and

$$p(n)^2 = (1-\beta)^2 \sum_{i=0}^{n-1}\sum_{j=0}^{n-1} \beta^i \beta^j e(n-i).e(n-i-1)e(n-j)e(n-j-1) \tag{4.21}$$

In the following analysis, the steady state performance of the proposed algorithm is studied. Therefore, assumption is made in analysis that the algorithm has converged. In this case, the samples of the error e(n) can be assumed uncorrelated. Using (4.13), (4.20), and (4.21), the mean and the mean-square behavior of the step-size, upon convergence, are

$$E(\mu(n+1)\} = \alpha E\{\mu(n) + \gamma(1-\beta)^2 \sum_{i=0}^{n-1} \beta^2 E\{e^2(n-i)\}.E\{e^2(n-i-1)\} \tag{4.22}$$

and

$$E\{\mu^2(n+1)\} \approx \alpha^2 E\{\mu^2(n)\} + 2\alpha\gamma E\{\mu(n)\}(1-\beta)^2.\sum_{i=0}^{n-1} \beta^{2i} E\{e^2(n-i)\}E\{e^2(n-i-1)\} \tag{4.23}$$

Following the same argument in [13], a sufficient condition that ensures convergence of the MSE is

$$0 < \frac{E\{\mu^2(\infty)\}}{E\{\mu(\infty)\}} \leq \frac{2}{3tr(R)} \tag{4.24}$$

where $E\{\mu(\infty)\}$, and $E\{\mu^2(\infty)\}$ are the steady-state values of $E\{\mu(n)\}$ and $E\{\mu^2(n)\}$.

The misadjustment is defined as [9]

$$M = \frac{\varepsilon_{ex}(\infty)}{\varepsilon_{\min}} \tag{4.25}$$

$$\frac{E\{\mu^2(\infty)\}}{E\{\mu(\infty)\}} \approx y \tag{4.26}$$

(Assuming that $\varepsilon_{ex}(\infty) << \varepsilon_{min}$)

where

$$y = \frac{2\gamma\varepsilon_{\min}{}^2(1-\beta)}{(1-\alpha^2)(1+\beta)} \tag{4.27}$$

From (4.24), (4.26) the following condition is imposed on γ, α and β to guarantee stability of the MSE.

$$0 < \frac{\gamma\varepsilon^2{}_{\min}(1-\beta)}{1-\alpha^2} \leq \frac{1}{3tr(R)} \tag{4.28}$$

The misadjustment is given by:

$$M \approx \frac{y}{2}tr(R) + \frac{N\sigma_n{}^2}{2E\{\mu(\infty)\}\varepsilon_{\min}} \tag{4.29}$$

In a stationary environment, σ^2_n =0, and the misadjustment is given by

$$M \approx \frac{y}{2}tr(R) \tag{4.30}$$

In a stationary environment, a large number of samples would be used in the estimation of $E\{e(n)e(n-1)\}$, i.e., the exponential weighting parameter β should be close to one. For the same level of misadjustment, a larger γ can be used while

maintaining the stability of the algorithm. A larger γ results in a larger step size in the initial stages of adaptation, i.e. faster convergence. Thus, the parameters γ and β provide the algorithm with an extra degree of freedom that facilitates better simultaneous control of both convergence speed and final excess MSE. In [13], the level of misadjustment of the VSS algorithm can be lowered by reducing the value of γ. However, this is achieved at the expense of slower convergence speed. When operating in a nonstationary environment, the choice of β becomes crucial. To provide good tracking capabilities, β should be small to cope with the time-varying statistics of the environment. The selection of β should achieve a compromise between tracking speed and excess MSE.

4.3.3 Simulation Results and Observations

As given in [14] the proposed modified variable step-size LMS (MVSS) algorithm is implemented for stationary and nonstationary environments in a system identification setup. The performance of the algorithm is compared with the variable step-size LMS (VSS) algorithm [13], the stochastic gradient algorithm with gradient adaptive step size (SGA-GAS) [25], and the fixed step-size LMS (FSS) algorithm [9]. Parameters of these algorithms are selected to produce a comparable level of misadjustment. The desired signal d(n) is corrupted by zero-mean, uncorrelated Gaussian noise of variance ε_{min}. Results are obtained by averaging over 200 independent runs [As reported in 14].

A. Example 1: White Input, Low SNR

The unknown moving average system has four time invariant coefficients, and the FIR adaptive filter is of equal order. Both are excited by a zero-mean, white Gaussian signal of unity variance. ε_{min} is equal to 1. The proposed MVSS is implemented with the parameters α=0.97 (which is found to be a good choice in stationary and nonstationary environments and as given in the paper) and β=0.99. For the other algorithms μ_{max}=0.1 and $\mu_{min}=1*10^{-5}$. In addition, $\mu = 3.5*10^{-4}$ for the FSS. fig. 4.1 shows that the MVSS algorithm provides the fastest speed of convergence among all other algorithms while retaining the same small level of misadjustment.

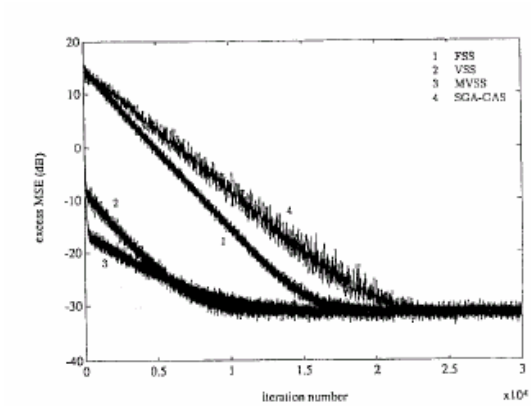


Fig 4.1 Comparison of Excess MSE of Various Adaptive Algorithms for White Input Case SNR = 0db

4.4 Gradient Adaptive Step Size LMS Algorithm

A variable step size algorithm proposed by Kwong and Johnston [13] can be used for acoustic echo cancellation but the results obtained are poor as compared to those for the other conventional adaptive algorithms. The variable step size algorithm, as appeared in [13] is of the form

$$w(n+1)=w(n)+\mu(n)x(n)e(n) \tag{4.31}$$

where $e(n)=d(n)-x(n)^T w(n)$ (4.32)

step size is updated as

$$\mu'(n+1) = \alpha\mu(n)+\beta e^2(n) \quad \text{with } 0<\alpha<1\ ,\ \beta>0 \tag{4.33}$$

and $\mu(n+1)= \mu_{max}$ if $\mu'(n+1)> \mu_{max}$

μ_{min} if $\mu'(n+1)< \mu_{min}$

$\mu'(n+1)$ otherwise

where $0<\mu_{min}<\mu_{max}$

To ensure stability, the variable step size μ (n) is constrained to the pre-determined maximum and minimum step size values of the LMS algorithm, while α and β are the parameters controlling the recursion. The results obtained by applying this algorithm for echo cancellation are poor. The average ERLE obtained is very low and the mean

square error is also quite large. Hence a modified gradient adaptive step size LMS algorithm has been proposed.

In the Gradient adaptive step size LMS adaptive algorithm the step size for each iteration is expressed as a vector, $\mu(n)$. Each element of the vector $\mu(n)$ is a different step size value corresponding to an element of the filter tap weight vector, $\mathbf{w}(n)$.

The gradient adaptive step size LMS algorithm (GLMS) is executed by following these steps for each iteration. Each iteration of the gradient LMS algorithm requires 4N+1 multiplication operations.

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

$$y(n) = w^T(n)\, x(n) \tag{4.34}$$

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

$$e(n) = d(n) - y(n) \tag{4.35}$$

3. The filter weight vectors, step size and the gradient g(n) are updated according to the following equations

$$g_i(n) = e(n)x(n-i) \tag{4.36}$$

for i= 0,1,2,……N-1

$$\mu_i(n) = \mu_i(n-1) + g_i(n)\, g_i(n-1) \tag{4.37}$$

$$\mu_i(n) = \begin{cases} \mu_{max} & \text{if } \mu_i(n) > \mu_{max} \\ \mu_{min} & \text{if } \mu i(n) < \mu_{min} \end{cases}$$

The performance of this algorithm is almost similar to the LMS algorithm and is better than the variable step size LMS algorithm discussed in [13].

Chapter 5

ECHO CANCELLATION-ANALYSIS AND OBSERVATIONS USING VARIOUS ADAPTIVE ALGORITHMS

5.1 Introduction

Echo is a natural phenomenon in long-distance telephone circuits because of amplification in both directions and series coupling of telephone transmitters and receivers at each end of the circuit. Echo suppressors, used to break the feedback, give one-way communication to the party speaking first. To avoid switching effects and to permit simultaneous two-way transmission of voice and data, adaptive echo cancellers are replacing echo suppressors worldwide. In broadband system architecture, two forms of echo cause problems for designers.

1. **Acoustic echo**: This form of echo is created by the acoustic properties of the areas where a VoP (voice over packet) phone is used. Acoustic echo occurs when an audio

signal is reverberated in a real environment, resulting in the original intended signal plus attenuated, time delayed images of this signal.

2. **Line echo**: This form of echo is electrical and is produced by a network hybrid (a 2-wire to 4-wire converter)

Echo cancellation is critical to achieving high quality voice transmissions over packet networks, which typically face transmission delays above 30 to 40 ms. These long delays make echo readily apparent to listeners, and must be eliminated in order to provide viable telephony service [16]. Echo cancellation is one of the most widely used digital signal processing devices in the world because each telephone call requires a pair of echo cancellers. Basically, a transversal filter, which is adaptively modeling the echo path impulse responses, generates an estimate of the echo, with this an echo estimate is created at the right time to cancel the actual echo. The common problems faced by echo cancellation are the convergence time and the degree of cancellation. Convergence time is the time taken to reach an acceptable level of steady state residual echo. Degree of cancellation is the amount of echo cancelled, measured in ERLE (echo return loss enhancement). Acoustic echo canceling is needed in various fields of communication to remove the coupling between the loudspeaker and the microphone; if not cancelled, this coupling results in an undesired acoustic echo that significantly degrades the sound quality. An acoustic echo canceller can overcome the acoustic feedback that interferes with teleconferencing and hands free telecommunication. It adaptively identifies the transfer function between a loudspeaker and a microphone, and then produces an echo replica that is subtracted from the real echo [7]

5.2 Acoustic Echo Cancellation

5.2.1 Origin of Acoustic Echo

Acoustic echo occurs when an audio signal is reverberated in a real environment, resulting in the original intended signal plus attenuated, time delayed images of this signal. The present analysis focuses on the occurrence of acoustic echo in telecommunication systems. Such a system consists of coupled acoustic input and output devices, both of which are active concurrently. An example of this is a 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.

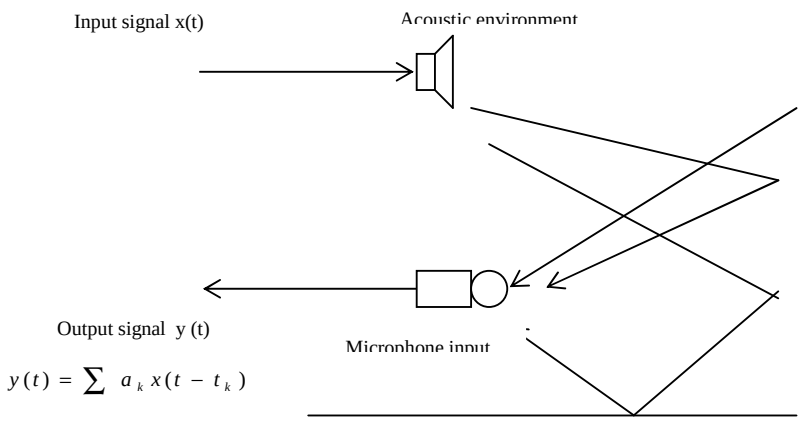


Fig 5.1 Origin of Acoustic Echo

These reverberated signals contain time-delayed images of the original signal, which are then returned to the original sender (Figure 5.1 a_k is the attenuation, t_k is time delay). The occurrence of acoustic echo in speech transmission causes signal interference and reduced quality of communication.

5.2.2 Hands Free Telephony

Echo cancellers have now been used for quite a long time. In the early days of telephony the need first arose since electrical echoes were generated in the splices between different line segments. These echoes made it practically impossible to use telephones since the echoes did propagate back and forth between the different splices with very little attenuation. To solve this problem line echo cancellers were constructed that removed the echoes by first predicting the echoes generated from a certain signal and then subtracting the predicted echo from the received signal. These echo cancellers were very successful and today almost no echo at all can be perceived while using telephones. In more recent years the need for echo cancellation has reappeared in a new area, namely hands-free telephony. The demand for hands-free

telephony is expected to increase enormously in the years to come. The principle for hands-free telephony is depicted in Figure 5.2.

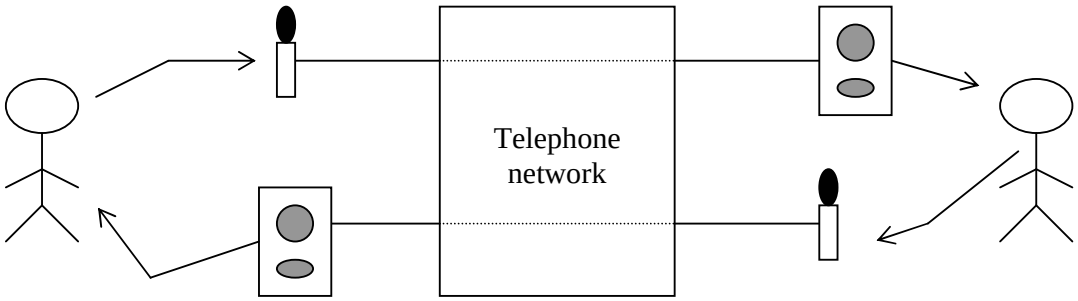


Fig 5.2 Hands Free Telephony

In hands-free telephony no telephone receiver is used. Instead a loudspeaker and a microphone is utilized. This setup has the undesired property that the sound emitted from the loudspeaker will be picked up by the microphone. Thus the speaker in the other end will hear an echo of his own voice when he speaks with the person with the hands-free telephony setup. If the delay in the echo is small this will not be noticed. This is the case when we talk in ordinary rooms and are not disturbed by the echoes from our own voices. In telephony, however, there is the additional delay for the signals to travel via the telephony network, and thus the echoes will be noticeable. Therefore, in some way the echoes from the loudspeaker picked up by the microphone must be removed from the microphone signal. That is the main principle and need of acoustic echo cancellation.

5.2.3 The Acoustic Echo Cancellation Problem

In almost all conversations echoes are present. Depending on the delay between the echoes and the echo-sources the echoes might be noticeable or not. If this delay exceeds a few tenths of millisecond echoes will be noticeable [28] and can be quite annoying. In hands-free telephony the echo delay is often that long due to the additional delay caused by the cables, microphones and speakers. A simple solution is to use only half-duplex communication, that is, block the communication in one

direction when the other direction is used. This works fine but is not desirable. What is desired is full duplex communication where communication in both directions is allowed simultaneously. To make this possible some kind of measure must be taken to eliminate the echo problem. This is where acoustic echo cancellation (AEC) is needed. To eliminate the echoes, a model of the echo path must be set up to allow estimation of echoes originating from different sounds. This model must be time varying since the echo path will be time varying (people walking in a room or the equipment is moved) and since the hands-free telephony setup will not be fixed (the microphone and loudspeaker might be positioned differently in different setups). The complexity of this model varies with the properties of the environment but to sufficiently attenuate the echoes it must be very high. Studies have shown that for an ordinary office room using a FIR filter about 2000 filter taps are required for a sampling frequency of 8 kHz. For a larger room with other echo characteristics such as a cathedral with stone walls, the required number of taps would be much larger. The adaptation of the time-varying model must naturally be performed in real time (since otherwise the users of the system would have to wait for the AEC to be performed and thus also have to wait for the sound as well). Due to the number of values to be adapted in the model of the echo path, this is quite a difficult task. Since the human ear can distinguish frequencies between 20 Hz and 20 kHz the sampling frequency should ideally be at least 40 kHz (two times the largest frequency component to avoid aliasing). This would require the filter to be updated 40000 times a second. Given that the computational requirements for the echo-estimation and updating of the filter are about 4000 multiplications and 2000 additions the result is that a total of 160.000.000 multiplication's and 240.000.000 additions must be performed every second. This is not feasible with today's technology and we are thus restricted to lower sampling rates such as 8 to 16 kHz. This is not a great limitation since normal human speech does not cover the whole frequency range audible to the human ear.

5.2.4 Configuration of an Acoustic Echo Canceller

The configuration of an acoustic Echo canceller is given in Fig 5.3. The echo canceller [7] identifies the transfer function of the acoustic echo path i.e. the impulse

response H(n) between the loudspeaker and the microphone. Since the impulse response varies as a person moves and varies with the environment, an adaptive filter is used to identify H(n). The desired signal is obtained by convolving the input signal with the impulse response of the acoustic environment and an echo replica is created at the output of the adaptive filter. The echo replica y(n) is then subtracted from the echo signal d(n) to give the error e(n)= d(n)-y(n).The adaptive FIR filter w(n) is adjusted to decrease the error power in every sampling interval. The adaptive algorithm should provide real time operation, fast convergence, and high echo return loss enhancement (ERLE). ERLE measures the amount of loss introduced by the adaptive filter alone. ERLE [5] is the ratio of send-in power and the power of a residual error signal immediately after the cancellation. It is measured in dB. ERLE depends on the size of the adaptive filter and the algorithm design. The higher the value of ERLE, the better the echo canceller.

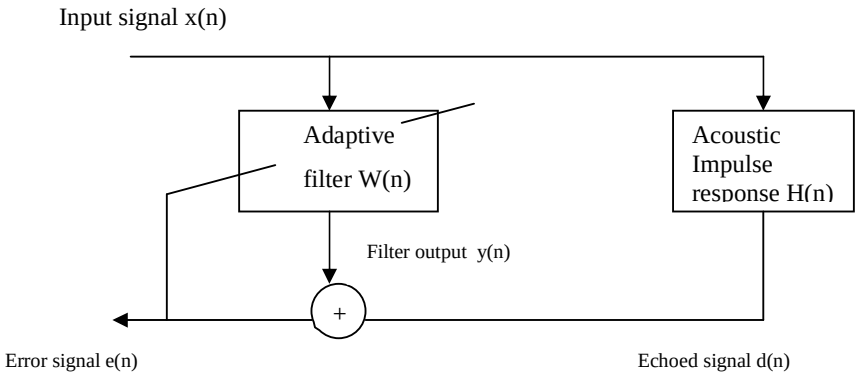


Fig 5.3 Configuration of an Acoustic Echo Canceller

5.3 Filter types for Acoustic Echo Cancellation

The two major filter types used for modeling the echo path are FIR filters and IIR filters.

5.3.1 IIR Filter

The IIR filter [1] is described by the difference equation (5.1)

(5.1)

which can be described by the transfer function (5.2),

$$H(z) = H_1(z)H_2(z) \tag{5.2}$$

where $H_1(z)$ and $H_2(z)$ are given by (5.3) and (5.4) respectively,

$$H_1(z) = \sum_{k=0}^{M} b_k z^{-k} \tag{5.3}$$

$$H_2(z) = \frac{1}{1 + \sum_{k=1}^{N} a_k z^{-k}} \tag{5.4}$$

The IIR filter can be realized in a structure called direct-form II, which is depicted in fig 5.4. The IIR filter is also called an AR (auto regressive) system when $H_1(z) = 1$. An advantage of the IIR filter is that it is good for filtering narrow frequency peaks since the transfer function has poles as well as zeros. Another advantage is that the impulse response of the IIR filter is infinite, thereby not requiring lots of filter parameters for implementing long impulse responses. The most significant drawback with the IIR filter is that it may be unstable (since it has poles).

5.3.2 The IIR filter in acoustic echo cancellation

At a first glance the IIR filter looks promising for echo cancellation since the long impulse response required for AEC is easily generated with this filter. Thus the number of parameters in the filter is not primarily determined by the impulse response length. A closer look reveals though that the number of parameters required is not much less than for the FIR filter. This is because the energy spectrum of an acoustic echo has lots of spectral peaks that are sharp. To successfully suppress these peaks the model has to include them. The IIR filter needs two parameters to model such a peak. Therefore the number of parameters required for the IIR filter is about the same as is required for the FIR filter. With IIR filters stability problems arise and the adaptation algorithms get more complicated and slower than for the FIR filter. For these reasons IIR filters are seldom used for echo cancellation. Instead FIR filters are used.

y (n)

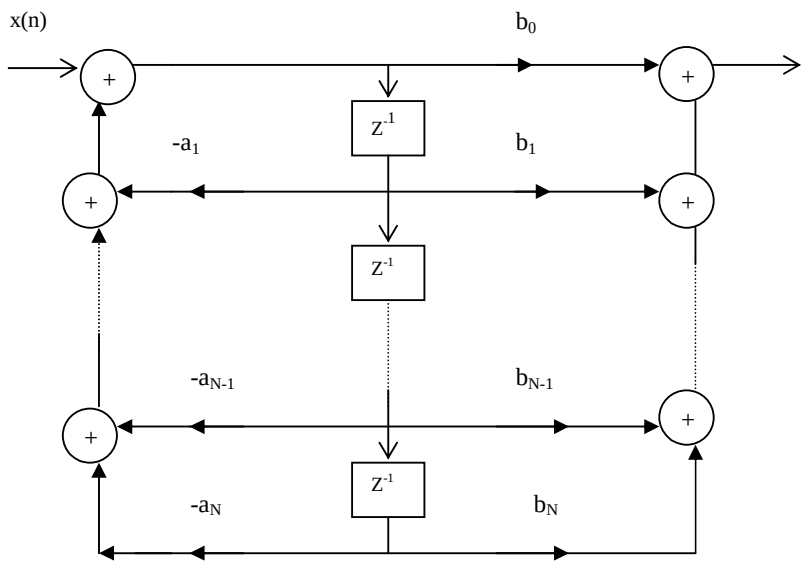


Fig 5.4 Direct Form II IIR Filter Realization For M=N

5.3.3 FIR Filter

A FIR filter [1] is characterized by the difference equation (5.5),

$$y(n) = \sum_{k=0}^{M-1} h_k x(n-k) \tag{5.5}$$

which has the transfer function (5.6),

$$H(z) = \sum_{k=0}^{M-1} h_k z^{-k} \tag{5.6}$$

The FIR filter can be realized using a direct-form structure (Figure 5.5). This realization is often called a tapped-delay-line filter where each filter coefficient constitutes a tap weight. The FIR filter is often called a MA (moving average) system since the output is a moving (weighted) average of the last M inputs.

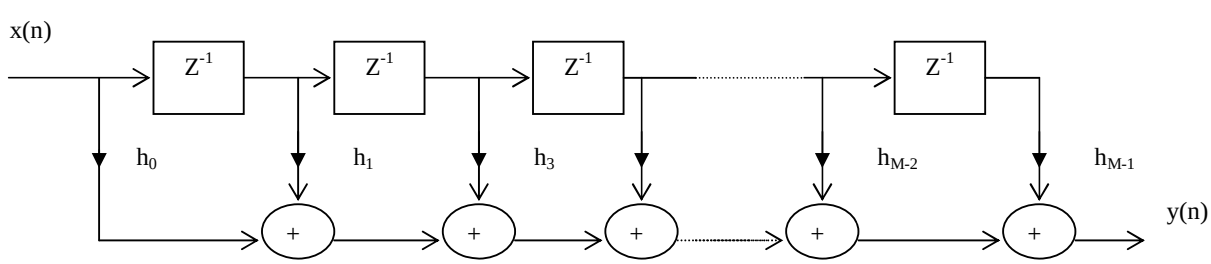


Fig 5.5 Direct Form FIR Filter Realization

The FIR filter has the advantage that it is inherently stable. It is also good for removing certain frequencies (since the transfer function has zeros for certain frequencies). A drawback is that the impulse response of the FIR filter is limited in time by the number of tap weights in the filter. This is a problem in acoustic echo cancellation where the echo path may be quite long, thus requiring a long impulse response which, with a FIR filter, only can be accomplished by a large number of tap weights. Another drawback is that it is bad for modeling narrow frequency bands since the transfer function has no poles.

5.3.4 The FIR Filter in the Acoustic Echo Cancellation

The property of the FIR filter that it is inherently stable is a desired one in AEC since no special action is then needed for ensuring stability of the filter. Thus FIR filters are the ones most often used for AEC. In AEC there is no true model of the system to compare with. Instead the amount with which the result of the echo cancellation differs from the desired result is measured. Good algorithms keep this small. In this measure all of the properties except the computational requirements are included. The computational requirements are critical for real-time echo cancellation. Since the algorithms are updated at the sampling-rate these must be fast enough to allow this. It is impossible to use algorithms for real-time echo cancellation that are too slow. Using faster hardware if such exists can of course solve this.

5.4 Echo Cancellation Using LMS Algorithm

The LMS algorithm has been discussed in section 3.2.1 in detail. In summary, we can write the LMS algorithm for every search iteration, in the form of three operations:

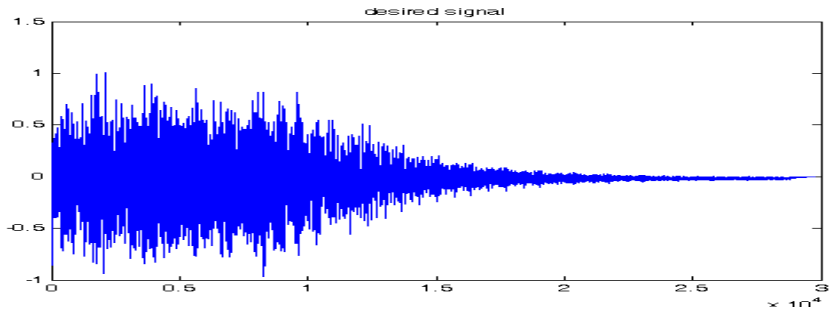
1. Filter output : $y(n) = w\, x(n)$ (5.7)
2. Error Estimation : $e(n) = d(n) - y(n)$ (5.8)
3. Tap Weight adaptation: $w(n+1) = w(n) + \mu\, e(n)x(n)$ (5.9)

The second operation defines the estimation error e(n), computed based on the current estimate of the tap weight vector w in the first operation. The last term in the third operation refers to the correction that is applied to the previous estimate of the tap weight vector. (Corresponding to step 3 of the method of steepest descent).

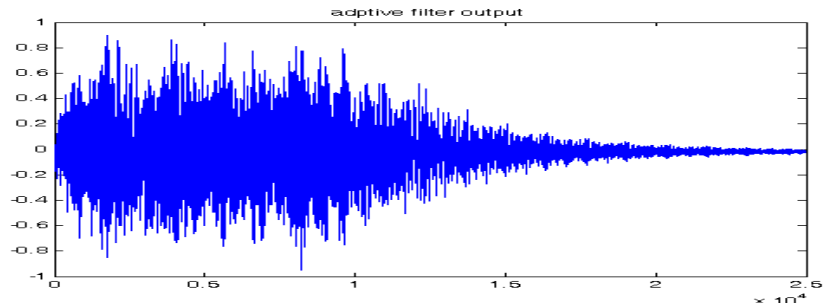
The desired signal has been obtained by convolving the input signal (vocal wav file) with the appropriately defined impulse response [30] of the acoustic environment. The plot for output of the adaptive filter shows that an estimate of the desired echo signal is obtained as its output and can thus be subtracted from the echo signal to minimize the error. The echo canceller [7] identifies the transfer function of the acoustic echo path i.e. the impulse response H(n) between the loudspeaker and the microphone. Since the impulse response varies as a person moves and varies with the environment, an adaptive filter is used to identify H(n). The desired signal is obtained by convolving the input signal with the impulse response of the acoustic environment and an echo replica is created at the output of the adaptive filter. The echo replica y(n) is then subtracted from the echo signal d(n) to give the error e(n)= d(n)-y(n). The simplicity and robustness of the LMS updating equation are the most important features and lead to many successful developments of other gradient descent-based algorithms. The LMS (least mean squares algorithm) is introduced as a way to recursively adjust the parameters of w(n) of a linear filter with the goal of minimizing the error between a given desired signal and the output of the linear filter.

5.4.1 Simulation Results for Echo Cancellation Using LMS Algorithm

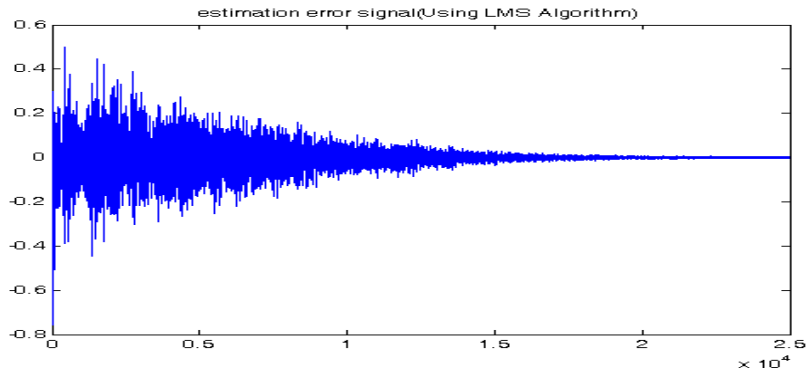
The following figures show the desired signal, adaptive filter output signal, estimation error, MSE and Echo return loss enhancement plots for the LMS algorithm with vocal input. The results have been obtained for 7500 iterations, 25000 iterations and with a filter length of 1000. The MSE shows that as the algorithm progresses the average value of the cost function decreases, this corresponds to the LMS filters impulse response converging to the actual impulse response, more accurately emulating the desired signal and thus more effectively canceling the echoed signal. The results have been shown only for no of iterations=25000 and filter order 1000.



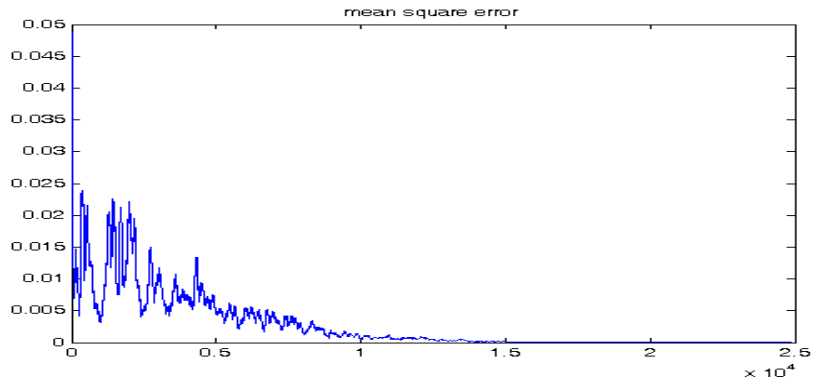
Plot of the desired signal i.e. the echo signal obtained by convolving the input signal with the appropriately defined impulse response.



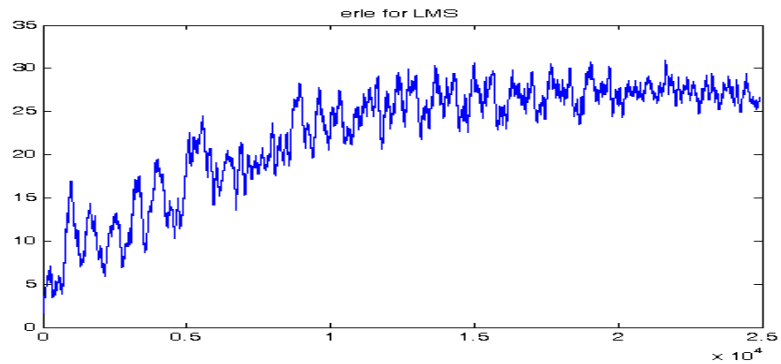
Plot of the adaptive filter output (shows that the output y(n) of the adaptive filter is an estimate of the echo signal)



Plot of the estimation error using LMS algorithm



Plot of the mean square error, which shows that the mean square error decreases as the algorithm progresses



Plot of ERLE for LMS algorithm. Average ERLE in dB obtained=15.34dB

Fig 5.6 Simulation Results for Echo Cancellation using LMS Algorithm (Number Of Iterations = 25000 And Filter Order 1000)

5.5 Echo cancellation using NLMS algorithm

The NLMS algorithm has been discussed in detail in section 3.2.7 .The NLMS algorithm is always the favorable choice because of fast convergence speed. For non-stationary input the correction term applied to the estimated tap weight vector w(n) at the n-th iteration is normalized with respect to squared Euclidean norm of the tap input x(n) at the (n-1)th iteration,

$$w(n+1) = w(n) + \frac{\alpha}{\|x(n)\|^2} e(n)x(n) \qquad (5.10)$$

where $\| \ \|^2$ = Euclidean Norm

When the input vector x(n) is small, instability may occur since we are trying to perform numerical division by small value of the Euclidean Norm

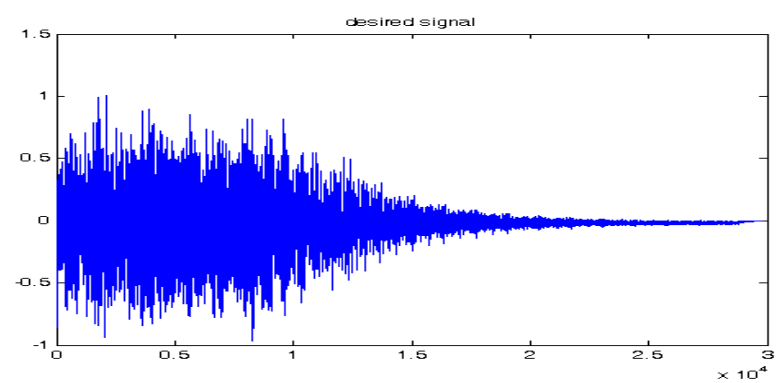
However, this can be easily overcome by appending a positive constant to the denominator in the above equation such that

$$w(n+1)=w(n) + \frac{\alpha}{c+\|x(n)\|^2} e(n)x(n) \qquad (5.11)$$

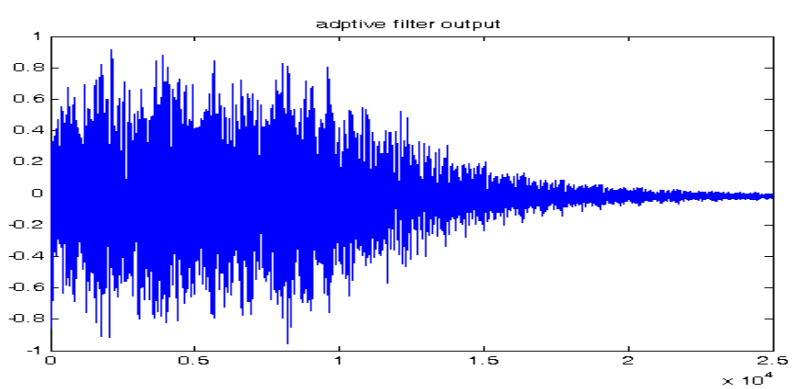
Where $c+\|x(n)\|^2$ is the normalization factor. With this, more robust and reliable implementation of the NLMS algorithm is obtained.

5.5.1 Simulation Results for Echo Cancellation Using NLMS Algorithm

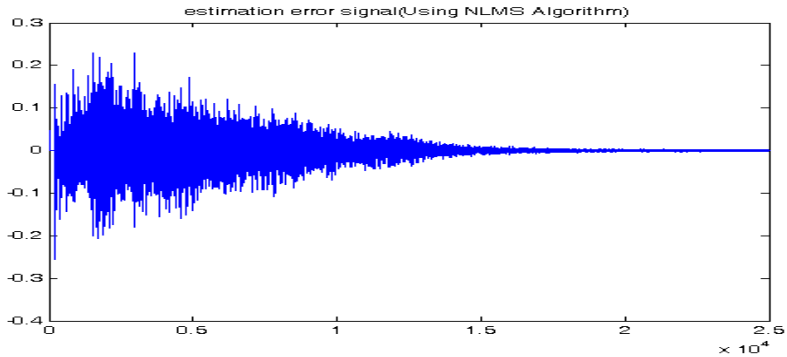
The results for the Normalized LMS algorithm are given below. The results obtained are better compared to that of the LMS, variable Step size LMS and Gradient adaptive step size algorithms. The estimation error and the mean square error are very small and the value of average ERLE is also quite large=22.1603dB. This is the best algorithm for the practical implementation of the echo canceller.



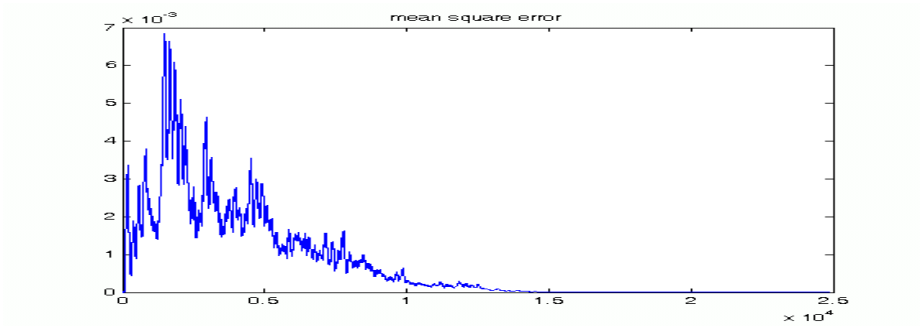
Plot for the desired signal for NLMS algorithm



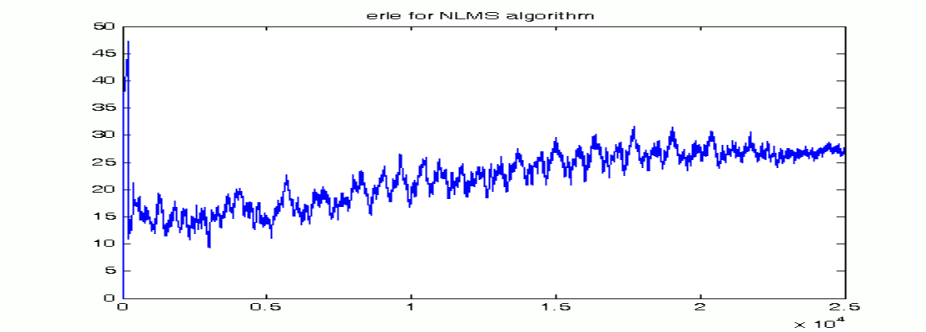
Plot of the adaptive filter output for NLMS algorithm



Plot for the estimation error for NLMS algorithm



Plot for the mean square error



Plot for the ERLE for NLMS algorithm (avg ERLE=22.16dB)

Fig 5.7 Simulation Results for Echo Cancellation using NLMS Algorithm (Number of Iterations = 25000 And Filter Order 1000)

Comparing this with fig 5.6, it can be seen that the error signal is smaller and the amplitude of the mean square error is in the order of ten times smaller than that of the standard LMS algorithm. This comes with the addition of only N more multiplication operations. The plot for the comparison of ERLE for LMS and NLMS algorithm is given below for 7500 and 25000 iterations:

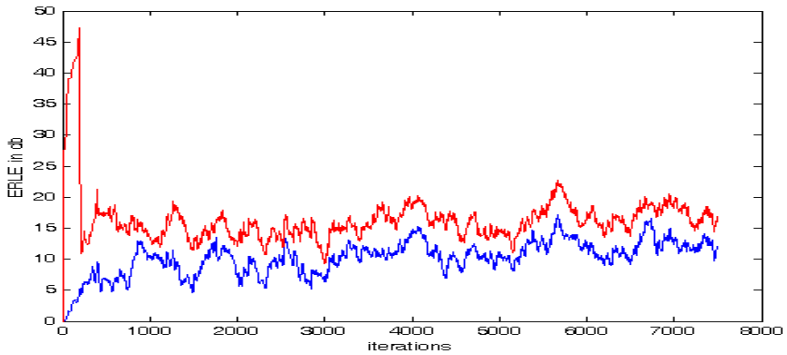


Fig 5.8 Comparison of the ERLE Plots for the LMS and NLMS Algorithm for 7500 Iterations

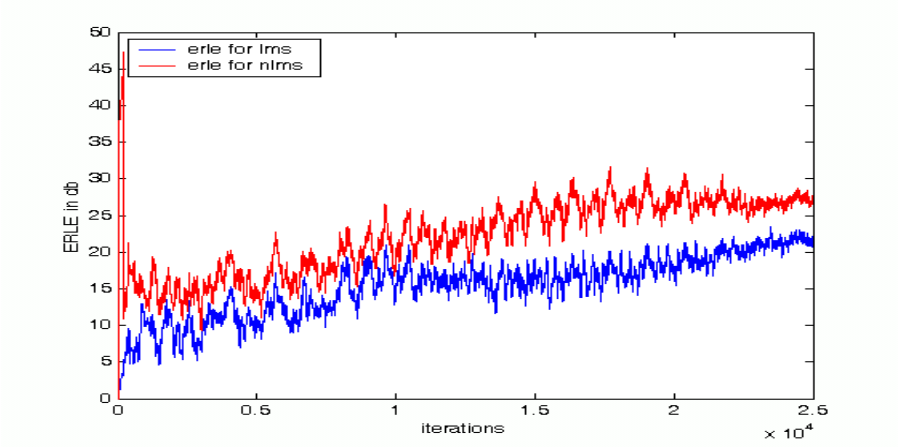


Fig 5.9 Comparison of the ERLE Plots for the LMS and NLMS Algorithm for 25000 Iterations

The curves look rather similar. If there is a difference, NLMS is superior, i.e.; its ERLE exceeds the ERLE of the LMS and grows quicker. The adaptive algorithm should provide real time operation, fast convergence, and high echo return loss enhancement (ERLE). NLMS has higher ERLE so it works as a better algorithm for echo cancellation application.

5.6 Echo Cancellation using RLS Algorithm

The equations for the implementation of the RLS algorithm are given below:

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

$$y'_{n-1}(n) = w'^{T}(n-1)x(n) \tag{5.12}$$

- The gain vector is calculated using the following equation:

$$k(n) = \frac{\lambda^{-1}\beta'^{-1}_{\lambda}(n-1)x(n)}{1+\lambda^{-1}x^{T}(n)\beta'^{-1}_{\lambda}(n-1)x(n)} \tag{5.13}$$

- The estimation error is calculated as

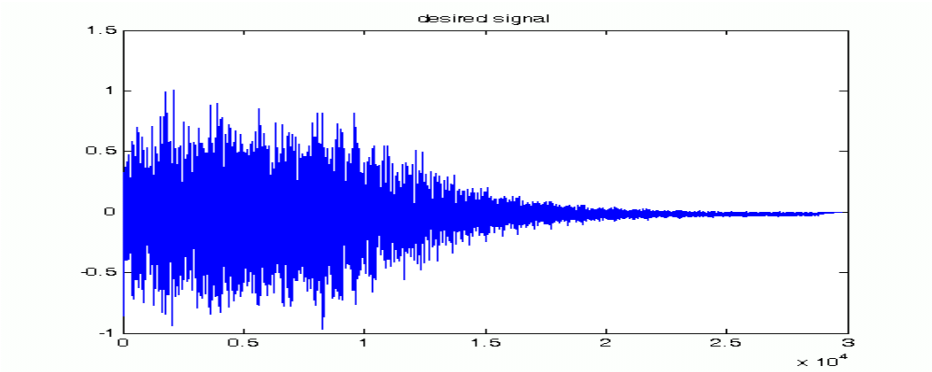
$$e'_{n-1}(n) = d(n) - y'_{n-1}(n) \tag{5.14}$$

- The filter tap weights can be calculated as

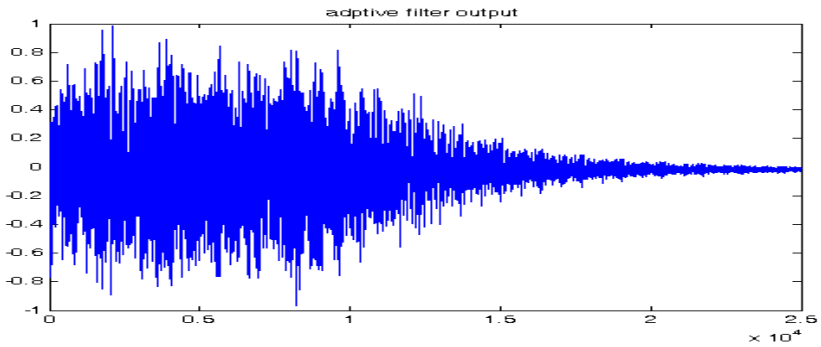
$$w'(n) = w'(n-1) + k(n)e'_{n-1}(n) \tag{5.15}$$

5.6.1 Simulation Results for Echo Cancellation using RLS Algorithm

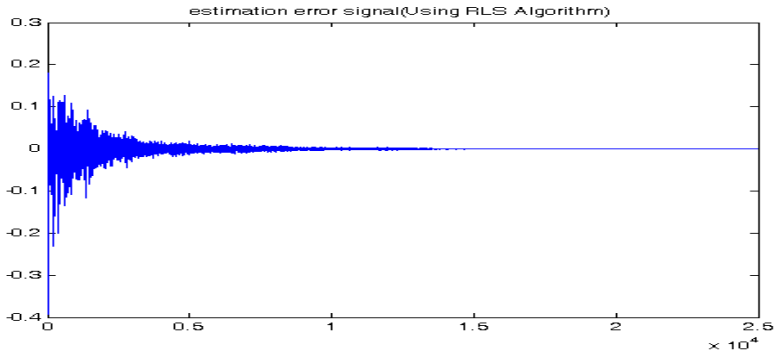
The results for the RLS algorithm are given below. From the plots it can be shown that the results obtained are the best for the RLS algorithm. The estimation error is very small, even smaller than the NLMS algorithm and the average ERLE is 40.2825dB, which is much higher than the LMS and the NLMS algorithm. Though the RLS algorithm gives much better results compared to other algorithms still it is not used, as each iteration requires $4N^2$ multiplications. For echo cancellation systems the FIR filter order is usually in the thousands. Thus the number of multiplications required are very large because of which the RLS algorithm is too costly to implement. In practice the LMS based algorithms, although poorer performers, are preferred.



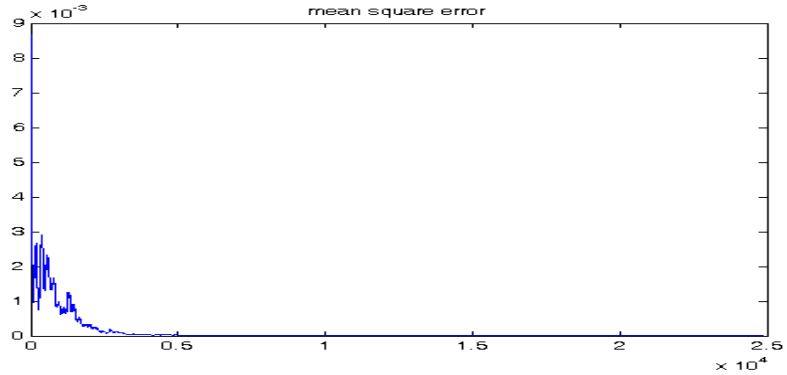
Plot for the desired signal for RLS algorithm



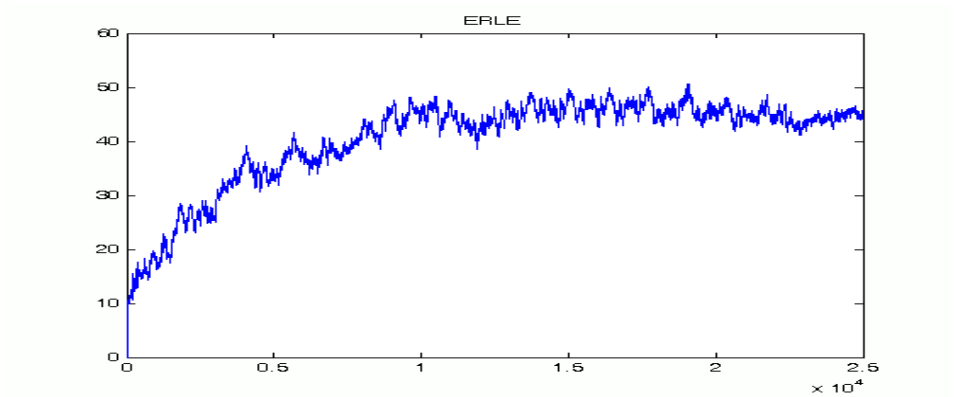
Plot for the adaptive filter output for RLS algorithm



Plot for the estimation error using RLS algorithm (much less as compared to the LMS algorithm)



Plot for the Mean square error using RLS algorithm(Several orders less than the LMS algorithm family)



Plot for ERLE for RLS algorithm (Average ERLE = 40.2825dB)

Fig 5.10 Simulation Results for Echo Cancellation using RLS Algorithm (Number Of Iterations = 25000 And Filter Order 1000)

5.7 Comparison of Convergence Rate of Various Algorithms

Convergence Rate: This quantity describes the transient behavior of the algorithm. This is defined as the number of iterations required for the algorithm to converge to its steady state mean square error or MSE. The steady state MSE is also known as the Mean asymptotic square error or MASE. This concerns how fast the algorithm will change the filter parameters to their final values.

The fig 5.11 shows that the Convergence Rate of NLMS algorithm is greater than LMS algorithm and the RLS algorithm has a far greater convergence rate compared to the LMS algorithm. Though the RLS algorithm gives much better results compared to other algorithms still it is not used, as each iteration requires $4N^2$ multiplications. For echo cancellation systems the FIR filter order is usually in thousands. Thus the number of multiplications required are very large because of which the RLS algorithm is too costly to implement. In practice the LMS based algorithms, although poorer performers, are preferred.

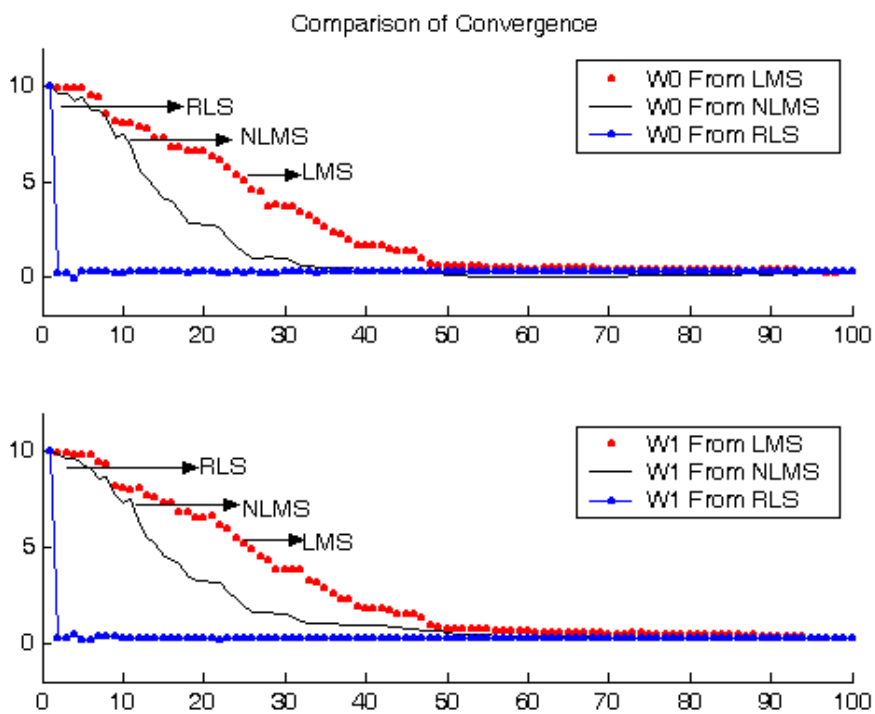


Fig 5.11 Comparison of Convergence Rate of Conventional Algorithms (LMS, NLMS and RLS Algorithms)

5.8 Echo Cancellation using VSS LMS Algorithm

The step size of the VSS LMS algorithm is adjusted as follows:

$$\mu(n+1) = \alpha\mu(n)+\beta e^2(n) \qquad (5.16)$$

To ensure stability, the variable step size μ (n) is constrained to the pre-determined maximum and minimum step size values of the LMS algorithm, while α and β are the

parameters controlling the recursion. Simply, the VSS algorithm's step size value change by tracking the error square or the error power.

The algorithm is of the form

$$w(n+1)=w(n)+\mu(n)x(n)e(n) \quad (5.17)$$

where $e(n)=d(n)-x(n)^T w(n)$ (5.18)

step size is updated as

$$\mu'(n+1) = \alpha\mu(n)+\beta e^2(n) \text{ with } 0<\alpha<1,\ \beta>0 \quad (5.19)$$

and $\mu(n+1)=\mu_{max}$ if $\mu'(n+1)>\mu_{max}$

μ_{min} if $\mu'(n+1)<\mu_{min}$

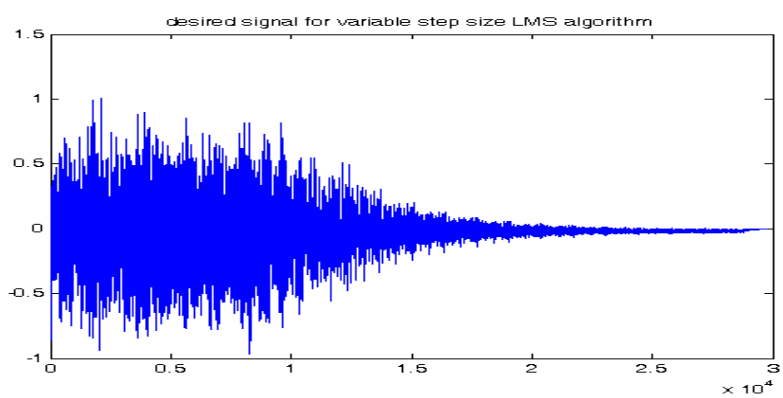
$\mu'(n+1)$ otherwise

where $0<\mu_{min}<\mu_{max}$

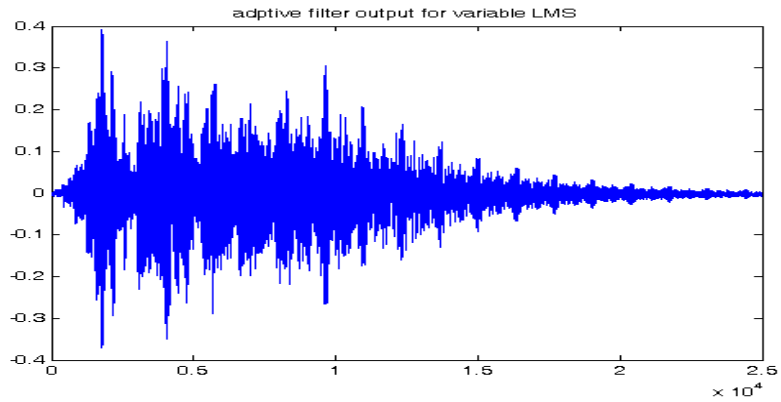
A typical value of α is chosen as given in the paper: $\alpha=0.97$, $\beta=4.8*10^{-4}$

5.8.1 Simulation Results for Echo Cancellation using VSS LMS Algorithm

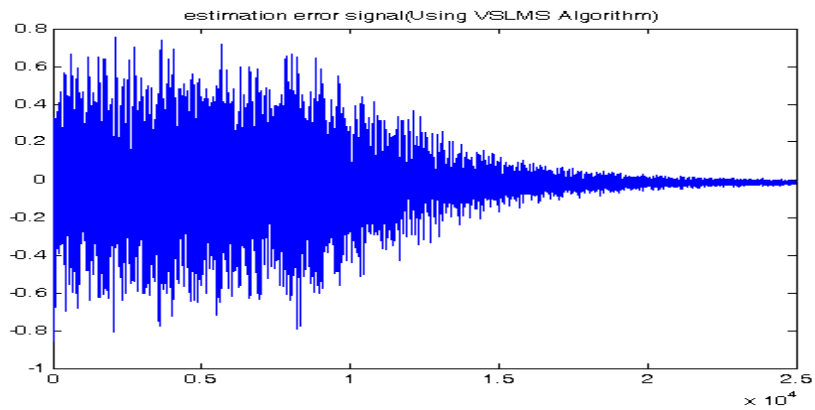
The results for the variable step size LMS algorithm given by above equations are given below. The results obtained are poor compared to the LMS algorithm. Error between the estimated signal and the desired signal is very large and the average value of ERLE is also quite small. Here max and min values of μ have been chosen as given in [13] i.e. $\mu_{max}=0.1$, and $\mu_{min}=10^{-5}$



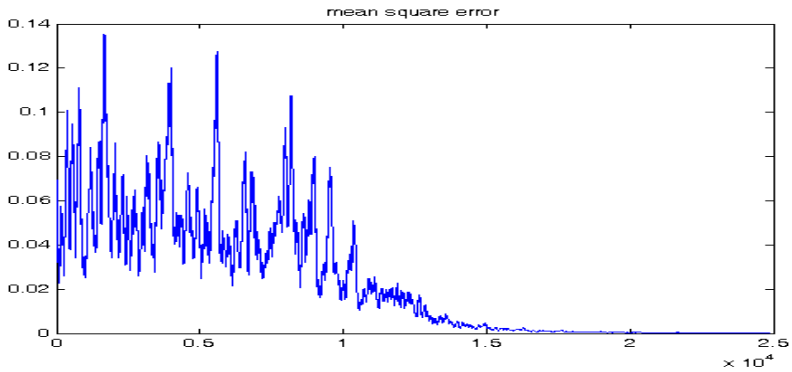
Plot of the desired signal for VSS algorithm



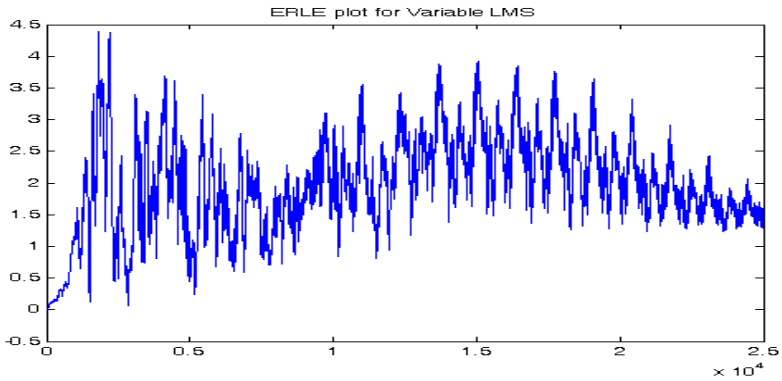
Plot of the adaptive filter output for VSS LMS algorithm



Plot of estimation error for VSS LMS algorithm



Plot of the Mean square error for VSS LMS algorithm



Plot for ERLE for VSS LMS algorithm (Average ERLE obtained = 5.9502 dB)

Fig 5.12 Simulation Results for Echo Cancellation using Variable Step Size LMS Algorithm (Number Of Iterations = 25000 And Filter Order 1000)

The fig 5.12 shows that the results obtained for the variable step size LMS algorithm are not satisfactory. The most important factor, which determines whether the algorithm is suitable to be used for echo cancellation i.e. the ERLE, is quite small. The value of average ERLE obtained is equal to 5.95dB, which is quite small as compared to the ideal value and that obtained for the other conventional algorithms. The Experimental results show that the performance of existing variable step size (VSS) algorithms is quite sensitive to the noise disturbance. Their advantageous performance over the LMS algorithm is generally attained only in a high signal-to-noise environment. This is intuitively obvious by noting that the criteria controlling the step-size update of these algorithms are directly obtained from the instantaneous error that is contaminated by the disturbance noise. Since measurement noise is a reality in any practical system, the usefulness of any adaptive algorithm is judged by its performance in the presence of this noise. The performance of the VSS algorithm deteriorates in the presence of measurement noise. Hence a new VSS LMS algorithm is proposed, where the step size of the algorithm is adjusted according to the square of the time-averaged estimate of the autocorrelation of e(n) and e(n-1) .

5.9 Echo cancellation using Modified VSS LMS algorithm

As seen above the performance of the VSS algorithm deteriorates in the presence of measurement noise. Hence a new VSS LMS algorithm is proposed, where the step size of the algorithm is adjusted according to the square of the time-averaged estimate of the autocorrelation of e(n) and e(n-1) . As a result, the algorithm can effectively adjust the step size as in [13] while maintaining the immunity against independent noise disturbance. The MVSS LMS algorithm allows more flexible control of misadjustment and convergence time without the need to compromise one for the other. The performance of the MVSS LMS algorithm is better that the VSS algorithm when applied to Echo cancellation application.

The weight update recursion is given by

$$W(n+1) = W(n) + \mu(n)e(n)X(n) \tag{5.20}$$

The estimate is a time average of e(n)e(n-1) that is described as

$$p(n) = \beta p(n-1) + (1-\beta)e(n)e(n-1) \tag{5.21}$$

Thus, the proposed step size update is given by

$$\mu(n+1) = \alpha\mu(n) + \gamma p(n)^2 \tag{5.22}$$

where limits on μ(n+1), α and γ are the same as those of the VSS LMS algorithm in [13]. The positive constant β (0< β <1) is an exponential weighting parameter that governs the averaging time constant, i.e., the quality of the estimation.

5.9.1 Simulation Results for Echo Cancellation using MVSS LMS Algorithm

As given in the paper [29], for the purpose of comparison, the VSS and the MVSS algorithms were used in an echo cancellation configuration. The echo path taken in the paper was simulated using the echo path impulse response of **CSA** loop ***#2*** from the set of HDSL2 test loops, at the central office end. The echo path impulse response was truncated to 124 *ps* to avoid the residual echo inherent in untruncated impulse

responses. The sampling interval was set at **1** ps, which resulted in an impulse response vector having a length of 124.

The echo path was excited with a zero-mean, white Gaussian signal to generate the echo. The echo signal thus generated was contaminated with the received signal, which was also a zero mean white Gaussian signal to result in an SNR of 0 dB.

The two algorithms were simulated with the following parameter values:

α=0.97, γ =0.08, β= 0.999

μ_{max}=0.01, and μ_{min}=0.001

To measure the performance of the two algorithms, the value of the misadjustment $||$ w* -w(n)$||$ where w* is the echo path impulse response vector and w(n) is the estimated coefficient vector of the LMS algorithm at the nth iteration, was calculated and plotted against the number of iterations in fig.5.13. As can be observed, the proposed algorithm yields a lower steady-state misadjustment.

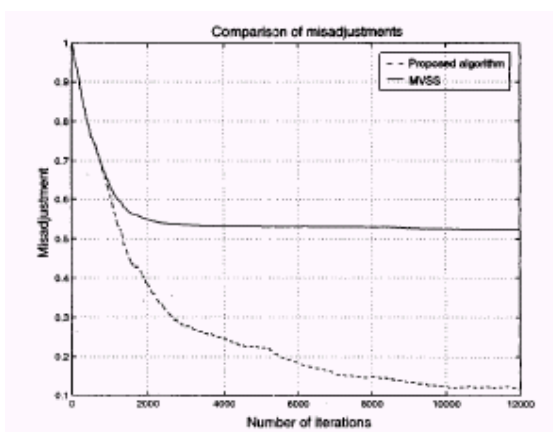
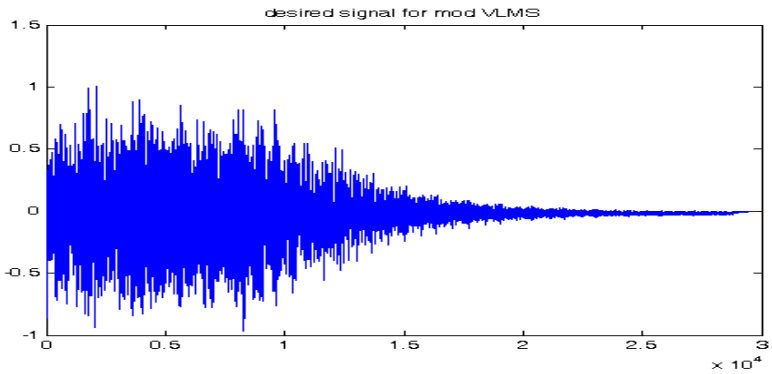
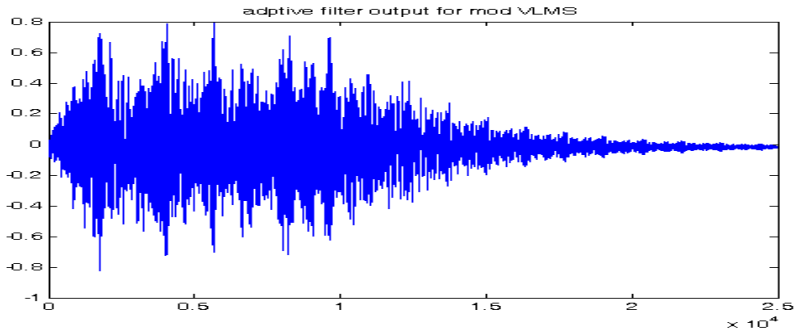


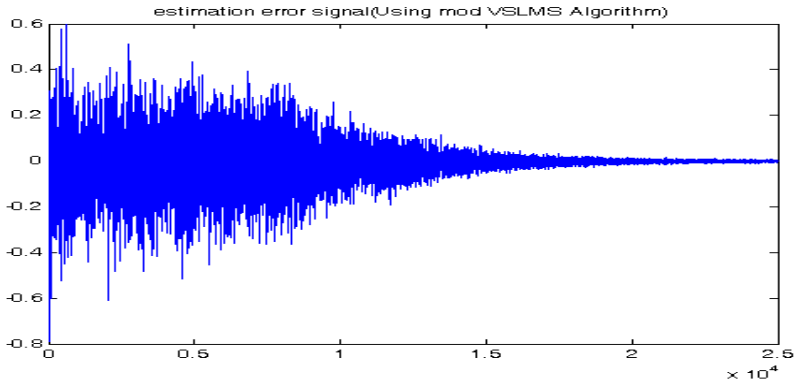
Fig 5.13: Comparison of the Misadjustments Using the MVSS and proposed VSS Algorithms.



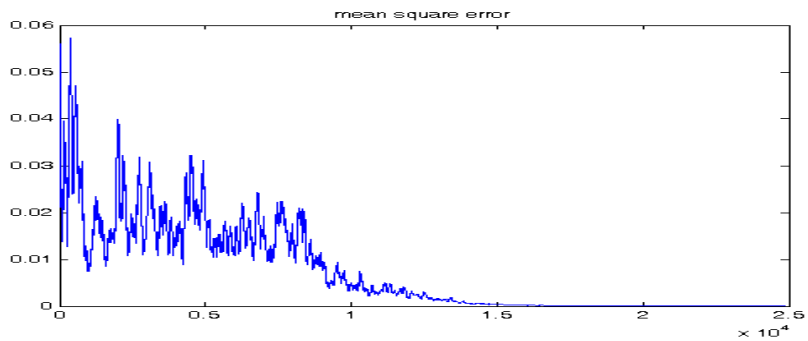
Plot of desired signal for Modified VSS-LMS algorithm



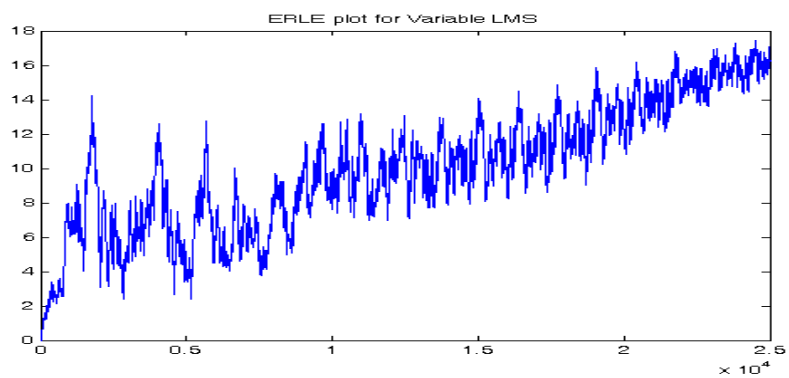
Plot of adaptive filter output for MVSS LMS algorithm



Plot of estimation error for MVSS LMS algorithm



Plot of mean square error for MVSS LMS algorithm



Plot of ERLE for MVSS LMS algorithm(average ERLE=15.56 dB)

Fig 5.14 Simulation Results for Echo Cancellation using Modified Variable Step Size LMS Algorithm (Number Of Iterations = 25000 And Filter Order 1000)

5.10 Echo Cancellation using Gradient Adaptive Step Size LMS Algorithm

Improved results in echo cancellation have been achieved with another variable step size algorithm i.e. by using the effect of gradient terms [33] on the update procedure.

The algorithm is as follows

$$w_i(n+1)=w_i(n)+2\,\mu_i g_i(n) \tag{5.23}$$

$$g_i(n) = e(n)x(n) \tag{5.24}$$

Where g is a vector comprised of the gradient terms, $g_i(n)=e(n)x(n-i)$, i=0….N-1, the length corresponds to the order of the adaptive filter.

$$\mu_i(n)= \mu_i(n-1)+\alpha sign(g_i(n))sign(g_i(n-1)) \quad (5.26)$$

For implementation

$g_i(n) = e(n)x(n-i)$

$g(n) = e(n)x(n)$

$\mu_i(n)= \mu_i(n-1)+\alpha g_i(n)g_i(n-1)$

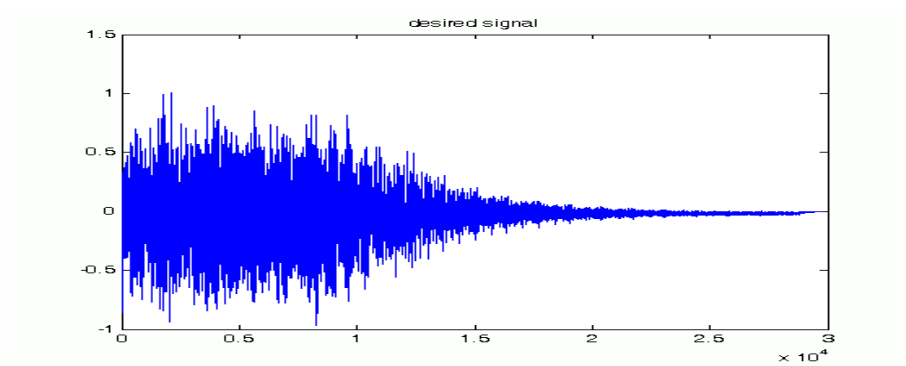
and if $\mu_i(n)> \mu_{max}$ if $\mu_i(n)= \mu_{max}$

if $\mu_i(n)< \mu_{min}$ if $\mu_i(n)= \mu_{min}$

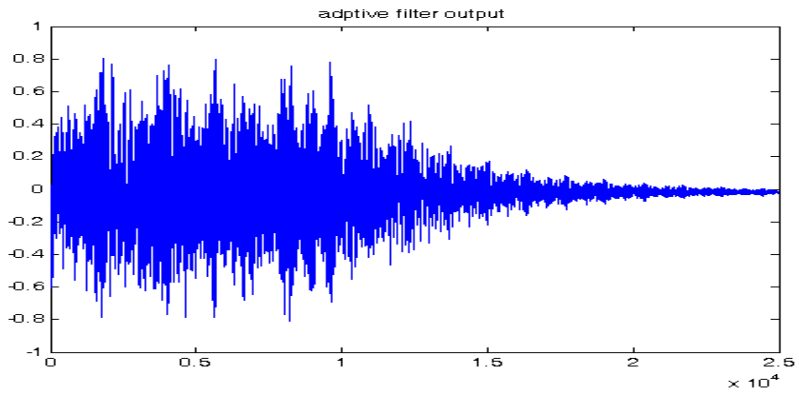
$w_i(n+1)=w_i(n)+2\ \mu_i(n)g_i(n)$

5.10.1 Simulation results for echo cancellation using GLMS algorithm

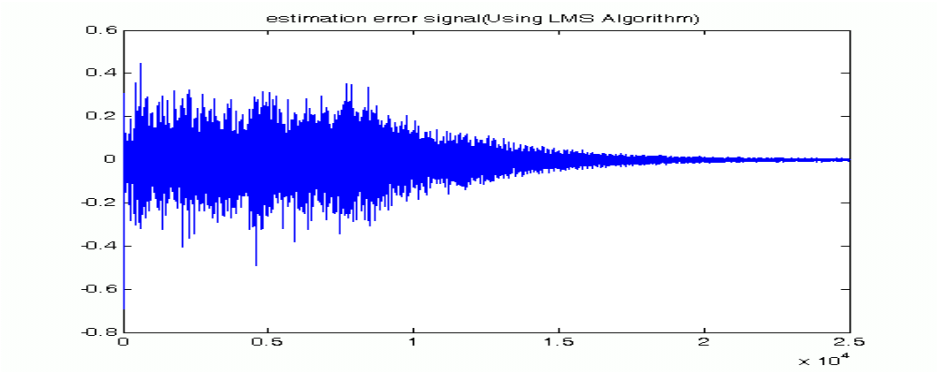
The results obtained for the gradient adaptive Step size LMS algorithm given by above equation are given below. The mean square error obtained in this case is about ten times compared to the MSE for LMS algorithm. Also the average ERLE is 11.6dB. Hence this shows a poorer performance compared to the LMS algorithm for echo cancellation but the performance is better compared to that obtained for the variable step size LMS algorithm. This also works as a modification of the existing variable step size LMS algorithm. Also the convergence rate for this is higher as compared to the LMS and the variable step size LMS algorithms.



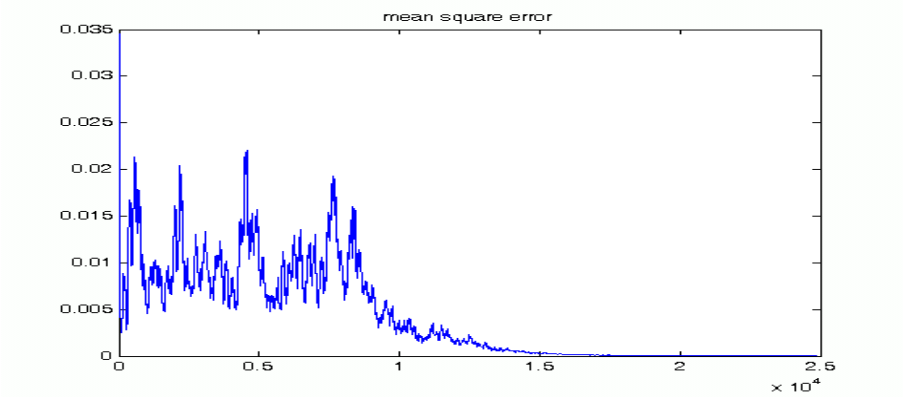
Plot of the desired (echo) signal for GLMS algorithm



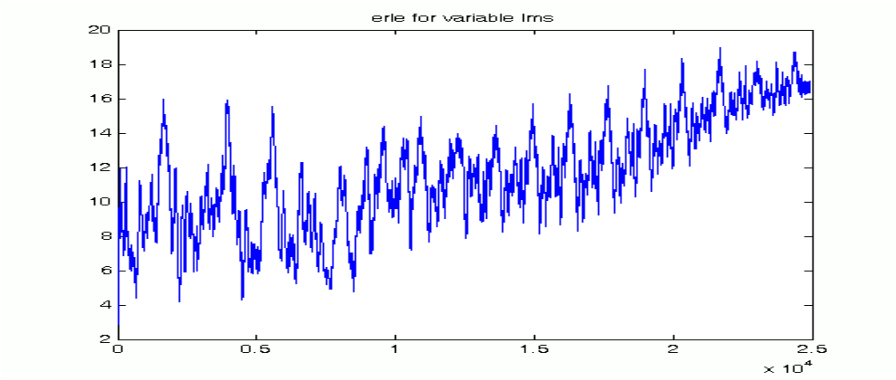
Plot for the adaptive filter output for GLMS algorithm



Plot of the estimation error for GLMS algorithm



Plot of the mean square Error



Plot of the ERLE for GLMS algorithm (average ERLE=11.6 dB)

Fig 5.15 Simulation Results for Echo Cancellation using Gradient Adaptive Step Size LMS Algorithm (Number of Iterations = 25000 And Filter Order 1000)

ALGORITHM	ITERATIONS	FILTER ORDER	MSE	AVG ERLE(in dB)	COMPUTATIONS
LMS	7500	1000	0.025	10.22	2N+1
NLMS	7500	1000	0.007	16.42	3N+1
RLS	7500	1000	0.003	29.64	$4N^2$
Variable step size	7500	1000	0.06	4.64	5N+1
MVSS LMS	7500	1000	0.03	11.6	2N+8
Gradient adaptive step size	7500	1000	0.02	9.1	4N+1

Table 5.1. Table for the comparison of performance of various algorithms used for ECHO cancellation (number of iterations=7500 and filter order=1000)

ALGORITHM	ITERATIONS	FILTER ORDER	MSE	AVG ERLE(in dB)	COMPUTATIONS
LMS	25000	1000	0.025	15.3489	2N+1
NLMS	25000	1000	0.007	22.1603	3N+1
RLS	25000	1000	0.003	40.2825	$4N^2$
Variable step size	25000	1000	0.12	5.9502	5N+1
MVSS LMS	25000	1000	0.03	15.56	2N+8
Gradient adaptive step size	25000	1000	0.02	11.6	4N+1

Table 5.2. Table for the comparison of performance of various algorithms used for ECHO cancellation (number of iterations=25000 and filter order N =1000)

Chapter 6

CONCLUSIONS AND FUTURE SCOPE

6.1 Conclusions

Adaptive Digital Signal Processing is a specialized branch of DSP, dealing with adaptive filters and system design. There are number of adaptive algorithms available in literature and every algorithm has its own properties, but aim of every algorithm is to achieve minimum mean square error at a higher rate of convergence with lesser complexity. Though many complex algorithms like RLS are having better performance but FSS – LMS is one of the most popular algorithms in the field of adaptive digital signal processing, due to simplicity. The VSS-LMS algorithm has been developed from FSS-LMS algorithm using one of its parameters.

The first part of the analysis emphasis on the basic adaptive algorithms and the modifications in VSS algorithm, which proved to be beneficial in adaptive system design. The discrete time version of Wiener filter theory has evolved from the pioneering work of Norbert Wiener on linear optimum filters for continuos-time signals. The importance of Wiener filter lies in the fact that it provides a frame of reference for linear filtering of stochastic signals, assuming the wide-sense stationarity.

At this point, it is informative to look at the FSS-LMS algorithm for stochastic inputs about the steepest descent algorithm. The aim, of both algorithms is to attain weight vector w_0 (optimum weight vector) but the steepest descent algorithm can achieve this in infinite number of iterations because of exact measurement of gradient vector at each iteration and it has a well defined learning curve, obtained by plotting the mean square error versus number of iterations because it consists of a sum of decaying exponentials, the number of which is equal to the number of tap coefficients. The correlation matrix R and cross correlation matrix p are computed through the use of ensemble averaging operation applied to the statistical populations of the tap inputs and the desired response. The FSS-LMS algorithm relies on the noisy estimate for the gradient vector. Therefore the mean square error is more in case of FSS-LMS algorithm and learning curve consists of noisy, decaying exponentials.

The convergence of the FSS-LMS algorithm in the mean square is assured by choosing the step size parameter in accordance with the practical condition. With small value of step size, the adaptation is slow, which is equivalent to large number of memory elements. The excess mean square error and misadjustment are less in this case because both are directly proportional to step size and number of tap elements. When value of step size is large, the adaptation is fast, giving rise to high mse and misadjustment.

The VSS-LMS algorithm has been developed from the FSS-LMS algorithm. Based on the error-squared power, Kwong and Johnston proposed a simpler Variable Step Size algorithm (VSS). The error power reflects the Convergence State of the adaptive filter, where a converging system has a higher error power while the converged system has a smaller error power. Therefore, scalar step size increases or decreases as the squared error increases or decreases, thereby allowing the adaptive filter to track changes in the system and produces a smaller steady state error.

Experimental result show that the performance of existing variable step size (VSS) algorithms is quite sensitive to noise disturbance. Their advantageous performance over the LMS algorithm is generally attained only in a high signal-to-noise environment. Since measurement noise is a reality in any practical system, the usefulness of any adaptive algorithm is judged by its performance in the presence of this noise. The performance of the VSS algorithm deteriorates in the presence of measurement noise. Hence a new VSS LMS algorithm is proposed , where the step size of the algorithm is adjusted according to the square of the time-averaged estimate of the autocorrelation of e(n) and e(n-1) . As a result, the algorithm can effectively adjust the step size as in while maintaining the immunity against independent noise disturbance. The MVSS LMS algorithm allows more flexible control of misadjustment and convergence time without the need to compromise one for the other.

In the second part of the thesis, adaptive algorithms have been applied to acoustic echo cancellation. The desired signal has been obtained by convolving the input signal (vocal wav file) with the appropriately defined impulse response of the acoustic environment. The tables 5.1 and 5.2 summarize the various results obtained and compare the performance of various algorithms used for Echo Cancellation. The

values of average ERLE obtained from the plots for various adaptive algorithms shows that the average ERLE is maximum for RLS algorithm. The estimation error and the mean square error are several orders smaller for RLS algorithm. The convergence rate of NLMS algorithm is greater than the LMS algorithm and the RLS algorithm has a far greater convergence rate compared to the LMS algorithm. Though the RLS algorithm gives much better results compared to other algorithms, still it is not used, as each iteration requires $4N^2$ multiplications. For echo cancellation systems the FIR filter order is usually in the thousands. Thus the number of multiplications required are very large because of which the RLS algorithm is too costly to implement. In practice the LMS based algorithms, although poorer performers, are preferred. Also the results obtained for Modified VSS-LMS algorithm are almost similar to the LMS algorithm but the Modified VSS-LMS algorithm can effectively adjust the step size while maintaining the immunity against independent noise disturbance. Also the MVSS LMS algorithm allows more flexible control of misadjustment and convergence time without the need to compromise one for the other.

Therefore, the overall analysis shows that the MVSS-LMS algorithm supersedes VSS-LMS adaptive algorithm under similar conditions, for different applications as far as misadjustment and rate of convergence is concerned.

6.2 Future Scope

Adaptive digital signal processing is a rapidly growing branch of DSP and has great significance in the design of adaptive systems. The various signal processing applications demand for reduction in trade off between misadjustment and convergence rate, taking realization of algorithm into account. There is a scope of improvement in existing algorithms, which reduces complexity and fulfills stringent conditions for stability.

This thesis 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 effective in an echo cancellation application.

The real time echo cancellation system can be implemented successfully using the TI TMSC6711 DSK. However, this system can only cancel those echoes that fall within the length of the adaptive FIR filter.

There's a lot, which can be done in future for improvement on the methods for acoustic echo cancellation The field of digital signal processing and in particular adaptive filtering is vast and further research and development in this area can result in some improvement on the methods studied in this thesis.

APPENDIX A

GAUSSIAN MOMENT FACTORING THEOREM

For zero mean Gaussian random variables x_i, i=1,…….,4 , the following result holds:

$$E(x_1x_2x_3x_4) = E(x_1x_2)E(x_3x_4) + E(x_1x_3)E(x_2x_4) + E(x_1x_4)E(x_2x_3)$$

Applying the above result to a zero mean Gaussian random variable X with $E(XX^T)=\Lambda$,

Λ = diag (λ_1, $\lambda_{2,.....,}$ λ_n),we obtain the following:

$$E(XX^TAXX^T) = \Lambda A\Lambda + \Lambda A^T\Lambda + \Lambda tr(\Lambda A)$$

If xi has mean $\overline{x}_i$, then on letting $x_i = \overline{x}_i$, we have

$$\begin{aligned} E(x_1x_2x_3x_4) = \; & E(\tilde{x}_1\tilde{x}_2)E(\tilde{x}_3\tilde{x}_4) + E(\tilde{x}_1\tilde{x}_3)E(\tilde{x}_2\tilde{x}_4) + E(\tilde{x}_1\tilde{x}_4)E(\tilde{x}_2\tilde{x}_3) + \\ & E(\tilde{x}_1\tilde{x}_2)\overline{x}_3\overline{x}_4 + E(\tilde{x}_1\tilde{x}_3)\overline{x}_2\overline{x}_4 + E(\tilde{x}_1\tilde{x}_4)\overline{x}_2\overline{x}_3 + E(\tilde{x}_2\tilde{x}_3)\overline{x}_1\overline{x}_4 \\ & + E(\tilde{x}_2\tilde{x}_4)\overline{x}_1\overline{x}_3 + E(\tilde{x}_3\tilde{x}_4)\overline{x}_1\overline{x}_2 + \overline{x}_1\overline{x}_2\overline{x}_3\overline{x}_4 \end{aligned}$$

APPENDIX B

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 [02].

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).

$$IfA = B^{-1} + CD^{-1}C^{H}$$
$$thenA^{-1} = B - BC(D + C^{H}BC)^{-1}C^{H}B$$

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

$$(A + \alpha aa^{T})^{-1} = A^{-1} - \frac{\alpha A^{-1}aa^{T}A^{-1}}{1 + \alpha a^{T}A^{-1}a}$$

References:

[1] John G. Proakis, Dimitris G.Manolakis, “Digital signal processing: principles, algorithms and applications”, Prentice Hall, March 2002.

[2] S. Haykins , “Adaptive Filter Theory” , Prentice Hall ,New Jersey, 1996.

[3] S. Qureshi, “Adaptive Equalization,” *Proceedings of the IEEE, vol.73*, No.9, pp. 1349-1387, Sept. 1985.

[4] A. Zerguine, C. F. N. Cowan, and M. Bettayeb, “ Adaptive Echo Cancellation Using Least Mean Mixed-Norm Algorithm”, *IEEE Trans. On Signal Processing* vol. 45, No.5, pp. 1340-1343, May, 1997.

[5] Hosien Asjadi, Mohammad Ababafha, “Adaptive Echo Cancellation Based On Third Order Cumulant, International Conference on Information, Communications and Signal Processing, ICICS '97 Singapore, September 1997

[6] D. G. Messerschmitt, “Echo Cancellation in Speech and Data Echo transmission,” *IEEE J. of Selected Areas in Comm*., *vol.* SAC-2, No.2, pp. 283-297, Mar. 1984

[7] Shoji Makino, Member, IEEE, Yutaka Kaneda, Member, IEEE and Nobuo Koizumi, “Exponentially weighted step size NLMS adaptive filter based on the statistics of a room impulse response”, *IEEE Trans. on speech and audio Processing*, vol. 1, No.1, pp.101-108, Jan 1993.

[8] Sudhakar Kalluri and Gonzalo R. Arce, “A General Class of Nonlinear Normalized Adaptive Filtering Algorithms”, *IEEE Trans. On Signal Processing* vol. 47, No.8, Aug, 1999

[9] B. Widrow, J. M. McCool, M. G. Larimore, and C. R. Johnson, Jr., “Stationary and non stationary learning characteristics of the LMS adaptive filter.” *Proc. IEEE.* vol. 64, pp. 1151-1162, 1976.

[10] B.Widrow, “Adaptive Noise Canceling”, *proc. IEEE*, vol.63, pp. 1692-1716, Dec1975.

[11] N.J. Bershad, “On the optimum gain parameter in the LMS adaptation”, *IEEE Trans. On Acoustic, Speech, Signal processing* vol. ASSP-35, pp.1065-1068, June 1981.

[12] S.Marcos and O.Macchi, "Tracking capability of the Least Mean square algorithm: Application of an asynchronous echo canceller", *IEEE Trans. On Acoustic, Speech, Signal processing* vol. ASSP-35, pp.1570-1578, Nov 1987.

[13] Raymond H. Kwong and Edward W. Johnston, "A Variable Step Size LMS Algorithm", *IEEE Trans. On Signal Processing* vol. 40, No.7, pp. 1633-1642, July, 1992

[14] Tyseer Aboulnasr, Member, IEEE, and K.Mayyas, "A robust variable step size LMS type algorithm: Analysis and Simulations", *IEEE Trans. On Signal Processing* vol. 45, No.3, pp. 631-639, March 1997.

[15] S.Theodoridis ,et. Al., "Efficient least square Adaptive algorithms for FIR transversal filtering", *IEEE Signal Processing Magazine,* vol. 16 , No.4, July,1999

[16] M.M. Sondhi, "An Adaptive Echo Canceller," *Bell Syst. Tech. J., vol. 46*, No.3, pp. 497-511, Mar. 1967

[17] B. Widrow, fellow, IEEE, J. M. McCool, senior member, IEEE, "**A** Comparison of Adaptive Algorithms Based on the Methods of Steepest Descent and Random Search", *IEEE transactions on antennas and propagation*, vol. AP-24, No.5, Sept 1976.

[18] B.Widrow, Mac.Cool , and M.ball, " The Complex LMS Algorithm", *proc.IEEE*,vol.63, pp. 719-720,1975.

[19] Geoffrey A. Williamson, Peter M. Clarkson, and William A. Sethares, "Performance Characteristics Of the Median LMS Adaptive Filter", *IEEE Trans. On Signal Processing* vol. 41, No.2, pp. 667-679, Feb 1993.

[20] Da-Zheng Feng, Xian-Da Zhang, Dong-Xia Chang, and Wei Xing Zheng , "A Fast Recursive Total Least Squares Algorithm for Adaptive FIR Filtering", *IEEE Trans. On Signal Processing* vol.52, No.10, pp.2729-2737, Oct 2004.

[21] Rabiner and Gold, "Theory And Applications Of Digital Signal Processing", Prentice Hall, Inc., 1975

[22] Wee-Peng Ang and B. Farhang-Boroujeny, "A New Class of Gradient Adaptive Step-Size LMS Algorithms", *IEEE Trans. on Signal Processing* vol. 49, No.4, April 2001.

[23] J. B. Evans, P. Xue, and B. Liu, “Analysis and implementation of variable step size adaptive algorithms,” *IEEE Trans. Signal Processing,* vol. 41, pp. 2517–2535, Aug. 1993.

[24] R. Harris *et al.*, “A variable step size (VSS) algorithm,” *IEEE Trans. Acoust., Speech, Signal Processing,* vol. ASSP-34, pp. 499–510, June 1986.

[25] V. J. Mathews and Z. Xie, “A stochastic gradient adaptive filter with gradient adaptive step size,” *IEEE Trans. Signal Processing,* vol. 41, pp. 2075–2087, June 1993.

[26] T. J. Shan and T. Kailaith, “Adaptive algorithms with an automatic gain control feature,” *IEEE Trans. Acoust., Speech, Signal Processing,* vol. 35, pp. 122–127, Jan. 1988.

[27] C.Breining, P.Dereiscitel, et al., “Acoustic echo control”, *IEEE signal processing Magazine,* Vol.16, no. 4, July, 1999

[28] Per Ahgren, “An environment for real time laboratory exercises in acoustic echo cancellation”, *Ph.D. Dissertation, Department of systems and control, Uppsala University, Uppsala, Sweden.*

[29] P.Sristi, W.S. Lu and A.Antoniou, “A new variable step size LMS algorithm and its applications in subband adaptive filtering for Echo Cancellation”, Department of Electrical and Computer Engineering ,University of Victoria, *IEEE Trans. Acoust., Speech, Signal Processing,*2001.

[30] Vincent Allen, Mark E Henderson, Erik S Wheeler, and John Williams, “Acoustic Echo Cancellation in a Reverberatory Chamber Using an Adaptive Least Means Square Algorithm, “The Echo Cancellation Group, Department of Electrical and Computer Engineering, Mississippi State University, Mississippi State, Mississippi

[31] Sundar G. Sankaran, “On ways to improve adaptive filter performance”, *Ph.D. Dissertation*, The Bradley Department of Electrical and Computer Engineering.

[32] Xiao Hu, Ai-Qun Hu, Yong Chen**,** Xiao-Hong Zeng , “An adaptive acoustic echo cancellation without double talk detection”, *Proceedings of the Second International Conference on Machine Learning and Cybernetics*, Xi’an, *2-5* November 2003.

[33] Farhang-Boroujeny, “Adaptive Filters, Theory and Applications” John Wiley and Sons, New York, 1999