10-15-2003

Implementation of an Acoustic Echo Canceller Using Matlab

Srinivasaprasath Raghavendran
University of South Florida

Implementation of an Acoustic Echo Canceller

Using Matlab

by

Srinivasaprasath Raghavendran

A thesis submitted in partial fulfillment
of the requirements for the degree of
Master of Science in Electrical Engineering
Department of Electrical Engineering
College of Engineering
University of South Florida

Major Professor: Wilfrido A. Moreno, Ph.D.
James T. Leffew, Ph.D.
Wei Qian, Ph.D.

Date of Approval:
October 15, 2003

Keywords: aec, nlms, dtd, nlp, matlab

ACKNOWLEDGEMENTS

I would like to express my sincere gratitude to my Major Professor Dr. Wilfrido A. Moreno, for being a constant source of help and inspiration throughout my work. His timely advice and guidelines have assisted me to get through a lot of difficult situations. My other committee members, Dr. James T. Leffew and Dr. Wei Qian have been very considerate and cooperative with me. I would like to thank them for their prompt feedback and being approachable and available whenever I needed any assistance. I would also like to thank the IEC forum for the help and valuable suggestions.

At this juncture, I thank my parents, my sister and my friends for their total support and encouragement. This Master’s thesis would not have been possible without their support.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

IMPLEMENTATION OF AN ACOUSTIC ECHO CANCELLER

USING MATLAB

Srinivasaprasath Raghavendran

ABSTRACT

The rapid growth of technology in recent decades has changed the whole dimension of communications. Today people are more interested in hands-free communication. In such a situation, the use a regular loudspeaker and a high-gain microphone, in place of a telephone receiver, might seem more appropriate. This would allow more than one person to participate in a conversation at the same time such as a teleconference environment. Another advantage is that it would allow the person to have both hands free and to move freely in the room. However, the presence of a large acoustic coupling between the loudspeaker and microphone would produce a loud echo that would make conversation difficult. Furthermore, the acoustic system could become instable, which would produce a loud howling noise to occur.

The solution to these problems is the elimination of the echo with an echo suppression or echo cancellation algorithm. The echo suppressor offers a simple but effective method to counter the echo problem. However, the echo suppressor possesses a main disadvantage since it supports only half-duplex communication. Half-duplex communication permits only one speaker to talk at a time. This drawback led to the

invention of echo cancellers. An important aspect of echo cancellers is that full-duplex communication can be maintained, which allows both speakers to talk at the same time.

This objective of this research was to produce an improved echo cancellation algorithm, which is capable of providing convincing results. The three basic components of an echo canceller are an adaptive filter, a doubletalk detector and a nonlinear processor. The adaptive filter creates a replica of the echo and subtracts it from the combination of the actual echo and the near-end signal. The doubletalk detector senses the doubletalk. Doubletalk occurs when both ends are talking, which stops the adaptive filter in order to avoid divergence. Finally, the nonlinear processor removes the residual echo from the error signal. Usually, a certain amount of speech is clipped in the final stage of nonlinear processing. In order to avoid clipping, a noise gate was used as a nonlinear processor in this research. The noise gate allowed a threshold value to be set and all signals below the threshold were removed. This action ensured that only residual echoes were removed in the final stage. To date, the real time implementation of echo an cancellation algorithm was performed by utilizing both a VLSI processor and a DSP processor. Since there has been a revolution in the field of personal computers, in recent years, this research attempted to implement the acoustic echo canceller algorithm on a natively running PC with the help of the MATLAB software.

CHAPTER 1

INTRODUCTION

1.1 Need for Echo Cancellation

In this new age of global communications, wireless phones are regarded as essential communications tools and have a direct impact on people's day-to-day personal and business communications. As new network infrastructures are implemented and competition between wireless carriers increases, digital wireless subscribers are becoming ever more critical of the service and voice quality they receive from network providers. Subscriber demand for enhanced voice quality over wireless networks has driven a new and key technology termed echo cancellation, which can provide near wire line voice quality across a wireless network.

Today's subscribers use speech quality as a standard for assessing the overall quality of a network. Regardless of whether or not the subscribers' opinion is subjective, it is the key to maintaining subscriber loyalty. For this reason, the effective removal of hybrid and acoustic echoes, which are inherent within the telecommunications network infrastructure, is the key to maintaining and improving the perceived voice quality of a call. Ultimately, the search for improved voice quality has led to intensive research into the area of echo cancellation. Such research is conducted with the aim of providing solutions that can reduce background noise and remove hybrid and acoustic echoes before any transcoder processing occurs. By employing echo cancellation technology, the quality of speech can be improved significantly. This chapter discusses the overall echo problem. A definition of echo precedes the discussion of the fundamentals of echo cancellation and the voice quality challenges encountered in today's networks.

1.2 Basics of Echo

Echo is a phenomenon where a delayed and distorted version of an original sound or electrical signal is reflected back to the source. With rare exceptions, conversations take place in the presence of echoes. Echoes of our speech are heard as they are reflected from the floor, walls and other neighboring objects. If a reflected wave arrives after a very short time of direct sound, it is considered as a spectral distortion or reverberation. However, when the leading edge of the reflected wave arrives a few tens of milliseconds after the direct sound, it is heard as a distinct echo [1].

Since the advent of telephony echoes have been a problem in communication networks. In particular, echoes can be generated electrically due to impedance mismatches at various points along the transmission medium. The most important factor in echoes is called end-to-end delay, which is also known as latency. Latency is the time between the generation of the sound at one end of the call and its reception at the other end. Round trip delay, which is the time taken to reflect an echo, is approximately twice the end-to-end delay.

Echoes become annoying when the round trip delay exceeds 30 ms. Such an echo is typically heard as a hollow sound. Echoes must be loud enough to be heard. Those less than thirty (30) decibels (dB) are unlikely to be noticed. However, when round trip delay exceeds 30 ms and echo strength exceeds 30 dB, echoes become steadily more disruptive. However, not all echoes reduce voice quality. In order for telephone conversations to sound natural, callers must be able to hear themselves speaking. For this reason, a short instantaneous echo, termed side tone, is deliberately inserted. The side tone is coupled with the caller's speech from the telephone mouthpiece to the earpiece so that the line sounds connected.

1.3 Types of Echo

In telecommunications networks there are two types of echo. One source for an echo is electrical and the other echo source is acoustic [1]. The electrical echo is due to the impedance mismatch at the hybrids of a Public Switched Telephony Network, (PSTN), exchange where the subscriber two-wire lines are connected to four-wire lines. If a communication is simply between two fixed telephones, then only the electrical echo occurs. However, the development of hands-free teleconferencing systems gave rise to another kind of echo known as an acoustic echo. The acoustic echo is due to the coupling between the loudspeaker and microphone. These electrical and acoustic echoes are discussed in greater detail in chapter 2.

1.4 The Process of Echo Cancellation

An echo canceller is basically a device that detects and removes the echo of the signal from the far end after it has echoed on the local end's equipment. In the case of circuit switched long distance networks, echo cancellers reside in the metropolitan

Central Offices that connect to the long distance network. These echo cancellers remove electrical echoes made noticeable by delay in the long distance network.

An echo canceller consists of three main functional components:

- Adaptive filter
- Doubletalk detector
- Non-linear processor

A brief overview of these components is presented in this chapter. However, a detailed sketch that involves mathematical illustrations is provided in chapter 3.

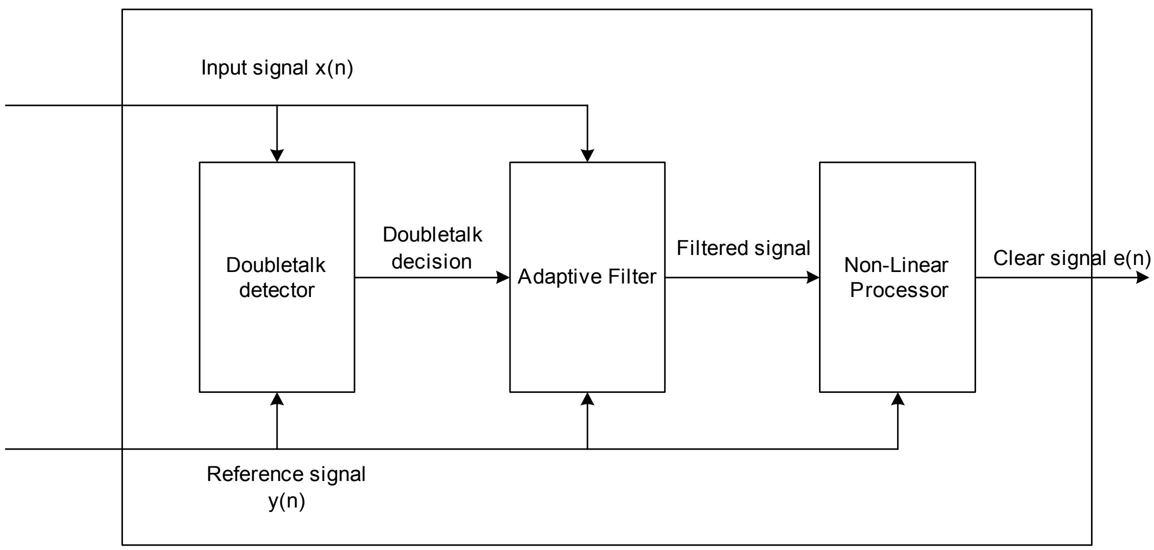

Figure 1.1: Block Diagram of a Generic Echo Canceller

1.4.1 Adaptive Filter

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

1.4.2 Doubletalk Detector

A doubletalk detector is used with an echo canceller to sense when far-end speech is corrupted by near-end speech. The role of this important function is to freeze adaptation of the model filter when near-end speech is present. This action prevents divergence of the adaptive algorithm.

1.4.3 Nonlinear Processor

The non-linear processor evaluates the residual echo, which is nothing but the amount of echo left over after the signal has passed through the adaptive filter. The nonlinear processor removes all signals below a certain threshold and replaces them with simulated background noise which sounds like the original background noise without the echo.

1.5 Echo Cancellation Challenges

An echo canceller has to deal with a number of challenges in order to perform robust echo cancellation.

1.5.1 Avoiding Divergence

The process of divergence is an adaptive filter problem that arises when a suitable solution for the line model is not found through the use of a mathematical algorithm. Under specific conditions, certain algorithms are bound to diverge and corrupt the signal or even add echo to the line. Good echo cancellers are tuned to avoid divergence situations in nearly all conditions.

1.5.2 Handling Doubletalk

In an active conversation, both talkers often speak at the same time or interrupt each other. Those situations are called "doubletalk". Doubletalk presents a special processing challenge to echo cancellers. Taken step-by-step, doubletalk proceeds as follows:

1. A speaks. The echo canceller must compare the received speech from Speaker A to what would be transmitted back to A in order to approximate an echo point.
2. B speaks over the echo signal. B speaking constitutes doubletalk. The echo canceller must detect the doubletalk and cancel the echo without affecting what is heard locally, which is speaker B's words.
3. The echo canceller must send B's speech, as well as the echo-cancelled version of A's own speech, back to A.

Handling doubletalk so that it sounds natural is technically challenging. A good echo canceller must be able to do the following:

- It must detect doubletalk and distinguish it from background noise.
- The echo canceller must be capable of choosing not to update the line model in order to avoid divergence if divergence could result.
- It needs to make a smooth transition between doubletalk detection, processing of doubletalk and return to the normal mode.

In summary, an important requirement for echo cancellation is the handling of doubletalk in a natural manner that does not cause divergence.

1.5.3 Preventing Clipping

Clipping occurs during a telephone conversation when part of the speech is erroneously removed. Clipping results due to the lack of a precise Non-Linear Processor, (NLP). Specifically, the NLP fails to start and stop at the right time. Typically, an NLP does not respond rapidly enough to the introduction of speech through the local end. It replaces parts of words with background noise, which makes the conversation hard to follow. The same can happen when the NLP confuses the fading of the voice level at the end of a sentence with a residual echo.

1.6 Research Motivation and Thesis Outline

Since echo cancellation is a very demanding process, real-time implementation has only been possible through the use of custom very large scale integration, (VLSI), processors or digital signal processors (DSP). These processors are specially designed for signal processing tasks. They provide parallel processing of commands and optimized pipeline structures. However, since the computation power of regular home personal computers, (PCs), has increased tremendously and powerful software has evolved, it is now possible to perform real-time signal processing in the PC environment as well. The advent of this growing capability was the motivation for this research. The objective of the research was the implementation of a software echo canceller running natively on a PC with the help of the MATLAB software.

This thesis provides an overview of an improved echo cancellation technique using a noise gate for the NLP. Chapter 1 discusses the definition of echo, the necessity of echo cancellers in telecommunications network, the basics of echo cancellation and the challenges of echo cancellation. Chapter 2 gives an overview of the types of echo and their sources. It also discusses, in great detail, the echo phenomena in four major telecommunication systems. The proposed echo cancellation algorithm is explained step-by-step in chapter 3. Chapter 4 discusses the simulation of the proposed algorithm, details of the simulation environment and the results obtained. Finally Chapter 5 provides a summary and some ideas concerning further work in this field.

CHAPTER 2

ECHOES IN TELECOMMUNICATION NETWORKS

This chapter deals with echoes that are generated in telecommunication systems. As discussed in chapter one, there are two main types of echo, which are termed electrical, or hybrid, and acoustic.

2.1 Hybrid/Electrical Echo

Hybrid echoes have been inherent within the telecommunications networks since the advent of the telephone. This echo is the result of impedance mismatches in the analog local loop. For example, this happens when mixed gauges of wires are used, or where there are unused taps and loading coils. In the Public Switched Telephone Network, (PSTN), by far the main source of electrical echo is the hybrid. This hybrid is a transformer located at a juncture that connects the two-wire local loop coming from a subscriber's premise to the four-wire trunk at the local telephone exchange. The four-wire trunks connect the local exchange to the long distance exchange. This situation is illustrated in Figure 2.1.

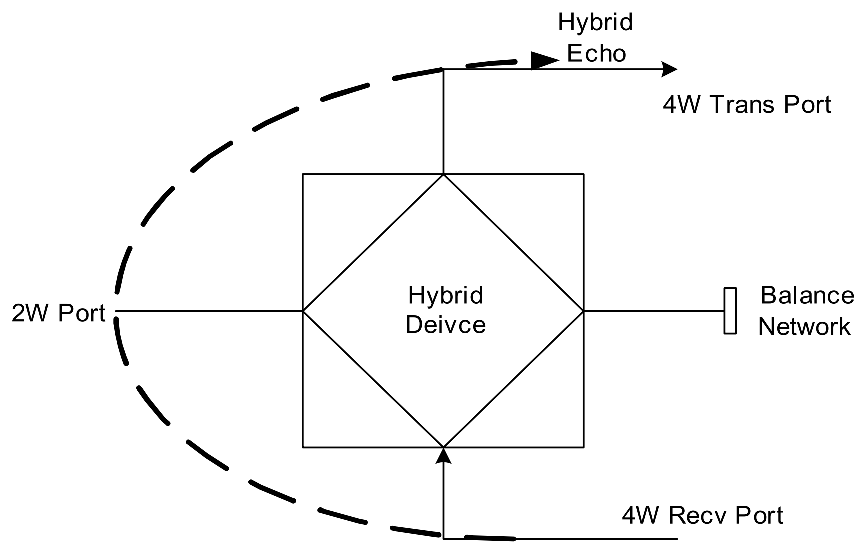

Figure 2.1: Hybrid Echo

The hybrid splits the two-wire local loop into two separate pairs of wires. One pair is used for the transmission path and the other for the receiver path. The hybrid passes on most of the signal. However, the impedance mismatch between the two-wire loop and the four-wire facility causes a small part of the received signal to "leak" back onto the transmission path. The speaker hears an echo because the far-end receives the signal and sends part of it back again. Electrical echo is definitely not a problem on local calls since the relatively short distances do not produce significant delays. However, the electrical echo must be controlled on long distance calls.

In the early years, when the public network was entirely circuit switched, the hybrid echo was the only significant source of echo. Since the locations of hybrids and most other causes of impedance differences in circuit switched networks were known, adequate echo control could be planned and provisioned. However, in today's digital networks the points where two wires split into four wires is typically also the point where analog to digital conversion takes place. Regardless of whether the hybrid and analog to digital conversion is implemented in the same device or in two devices, the two to four wire conversions constitute an impedance mismatch and echoes are produced [1].

2.2 Acoustic Echo

The acoustic echo, which is also known as a "multipath echo", is produced by poor voice coupling between the earpiece and microphone in handsets and hands-free devices. Further voice degradation is caused as voice-compressing and encoding/decoding devices process the voice paths within the handsets and in wireless networks. This results in returned echo signals with highly variable properties. When compounded with inherent digital transmission delays, call quality is greatly diminished for the wireline caller.

Acoustic coupling is due to the reflection of the loudspeaker's sound waves from walls, door, ceiling, windows and other objects back to the microphone. The result of the reflections is the creation of a multipath echo and multiple harmonics of echoes, which are transmitted back to the far-end and are heard by the talker as an echo unless eliminated. Adaptive cancellation of such acoustic echoes has become very important in hands-free communication systems such as teleconference or videoconference systems [1]. The multipath echo phenomenon is illustrated in Figure 2.2.

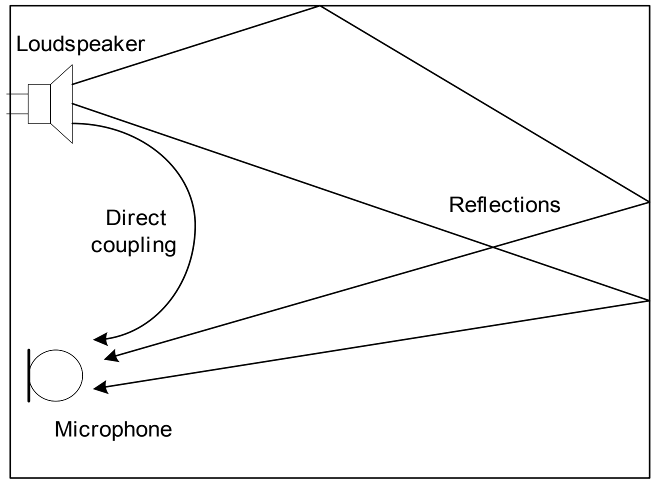

Figure 2.2: Sources of Acoustic Echo in a Room

In the following sections, the echo phenomena of four communication systems will be described. The communication systems are:

- Long-distance connections between fixed telephones
- Full-duplex data transmission between voice-band modems
- Short-distance connections between fixed and cellular telephones
- Teleconference/videoconference systems

2.3 Long Distance Calls between Fixed Telephones

A simple long-distance telephone connection is presented in Figure 2.3. This connection contains two-wire sections at the ends, the subscriber loops and possibly some portion of the local network. It also contains a four-wire section in the center, which is a carrier system for medium-range to long-range transmissions.

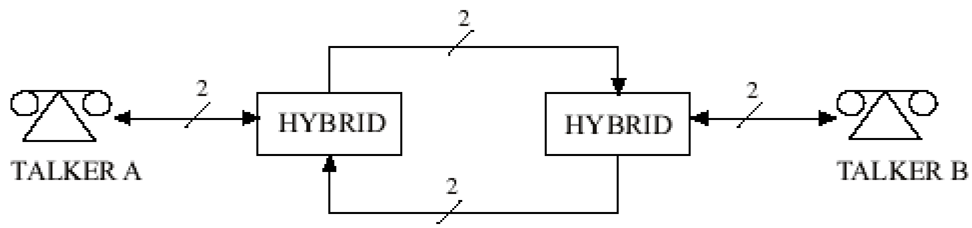

Figure 2.3: Simplified Long Distance Connections

Every conventional telephone in a given geographical area is connected to the local PSTN exchange by a two-wire line, called the subscriber loop, which carries a connection for both directions of transmission. Simply connecting the two subscriber loops at the local exchange sets up a local call. However, amplification of the speech signal becomes necessary when the distance between the two telephones exceeds 35 miles. Therefore, a four-wire line is required, which segregates the two directions of transmission. A hybrid is used to convert from the two-wire to four-wire line and vice versa.

An echo can be decreased if the hybrid has a significant loss between its two four-wire ports. To achieve this large loss the hybrid has to be perfectly balanced by impedance located at its four-wire portion. Unfortunately, this is not possible in practice since it requires knowledge of the two-wire impedance, which varies considerably over the population of subscriber loops. When the bridge is not perfectly balanced, impedance mismatch occurs. This causes some of the talker's signal energy to be reflected back as an echo. Adding an insertion loss to the four-wire portions of the connection can control the effects of echo. Such action is effective since the echo signals experience this loss two or three times while the talker's speech suffers this loss only once. However, on long-range connections the insertion loss can become very significant. Hence, it is not a favorable solution and other echo control techniques such as echo suppression must be used [1].

2.3.1 Echo Suppressors

Echo suppressors have been used since the introduction of long distance communication. This device basically takes advantage of the fact that people seldom talk simultaneously. The situation of two people talking simultaneously is termed “double talking”. The echo suppressor is also helped by the fact that during such double talking poor transmission quality is less noticeable. Figure 2.4 illustrates how the echo suppressor dynamically controls the connection based on who is talking, which is decided by the speech and double talking detector. Double talking is detected if the level of the signal in path L1 is significantly lower than that in path L2. When the far-end talker A is speaking, the path used to transmit the near-end speech is opened so that the echo is prevented. Then, when the near-end talker B speaks, the same switch is closed and a symmetric one at the far-end talker A’s path is opened. However, echo suppressors can clip speech sounds and introduce impairing interruption. For example, if talker B is initially listening to talker A but suddenly wants to talk, it is quite likely that the switch preventing talker A’s echo from being transmitted will not close quickly enough. This will cause the far-end talker A to not be able to receive all the messages from the near-end talker B. This deletion is noticed by talker A, encouraging him/her to stop and wait for talker B to finish. The resulting confusion may stop the conversation entirely while each party waits for the other to say something [1]. Therefore the best solution for removing echoes is to use echo cancellers. Echo cancellers are described in chapter 3.

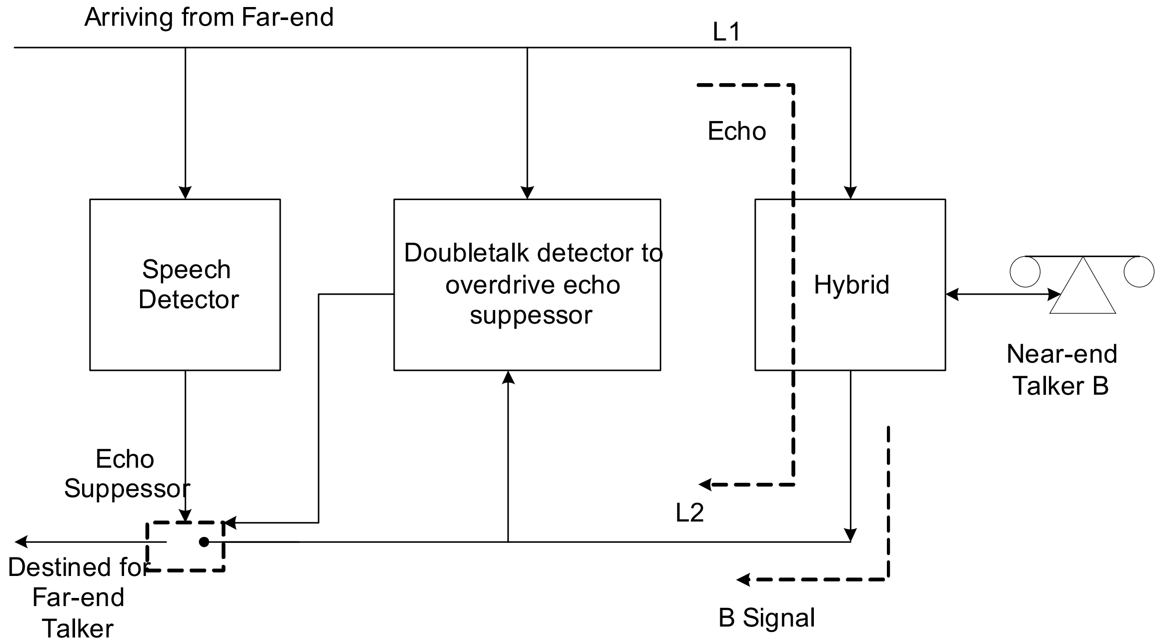

Figure 2.4: Echo Suppressor at Near-end Talker B Path

2.4 Full-duplex Data Transmission between Voice-band Modems

The two-wire telephone line of a subscriber loop can be used for the transmission of data through a modem. This can be accomplished either by using the entire bandwidth of the wire or transmitting the data on a bandwidth that is slightly above the one used to carry the speech signal. On an analog subscriber loop the speech signal occupies the bandwidth between 300 to 3400 Hz. A higher bit rate of up to 16 kbps can be transmitted by modulating the data signal onto a carrier signal at a band above 4000 Hz. Echo cancellation is needed for full-duplex communication within the same bandwidth over the subscriber loop as shown in Figure 2.5 where EC is the echo canceller, H is the hybrid, RX is the receiver and TX is the transmitter.

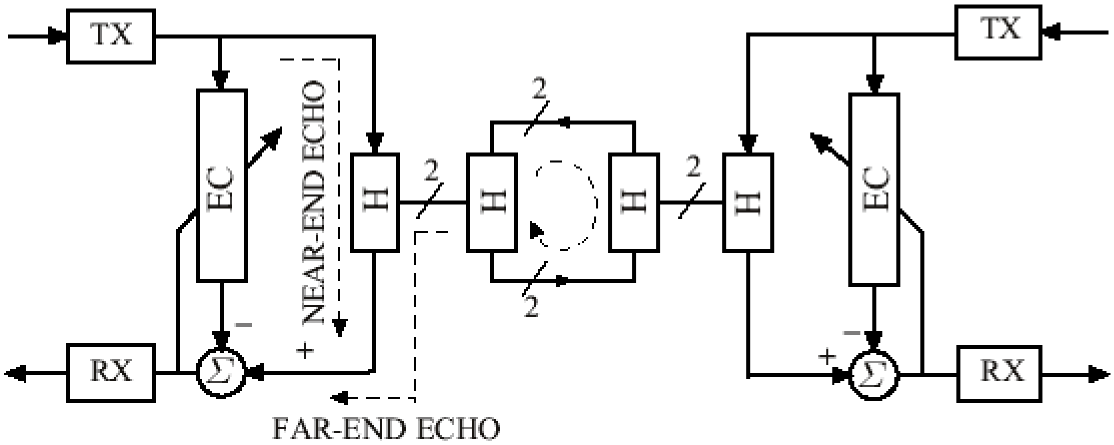

Figure 2.5: Echo Cancellers at Modem Locations for Full-Duplex Voice-band Modems

Typically the echo cancellers must be placed at the line interface where the hybrids connect the modem to the two-wire subscriber loop. Several problems are associated with this type of application and some of them are given below.

- It is not practical to freeze the adaptation algorithm during doubletalking in the case of full-duplex operation since the echo path's characteristic is likely to change during a lengthy communication session.
- The far-end echo, which is returned from the far-end hybrid, must also be taken into account. Therefore, the entire echo delay becomes very large, which is unique to the echo cancellation at the station, or modem, location. If the circuit includes a satellite communication network's four-wire link, the far-end echo will be delayed for more than 500ms. In such a case two cancellers will be required. One for the near-end and one for the far-end echo at the modems.
- A significantly high level of echo cancellation is required. The data signal coming from a far-end modem may be attenuated by 40 to 50dB. Therefore, the near-end echo, which is returned from the first hybrid at the local station, can be

40 to 50dB higher than the desired signal. For reliable communication the echo canceller must be able to attenuate the near-end echo by 50 to 60dB in order to maintain the signal power approximately 10dB above the echo [2].

2.5 Short Distance Connections between Fixed and Cellular Lines

In digital cellular communication, the combination of channel coding, speech coding and signal processing involves considerable delays. In most cases, the delays are increased further by time division multiple access framing. The total one-way delay can be from 30 to 120 msec. Figure 2.6 illustrates that only one echo canceller, (EC), facing the local PSTN exchange, (LE), is required in a digital cellular to fixed telephone connection. This is only possible if the cellular telephone is assumed to behave in a perfect four-wire fashion with no significant acoustic cross talk echo between the microphone and the earpiece of the cellular phone. However, under certain conditions, the cross talk echo in cellular handsets is still noticeable by users. Hence, the echo needs to be removed by cellular cross talk control devices [2].

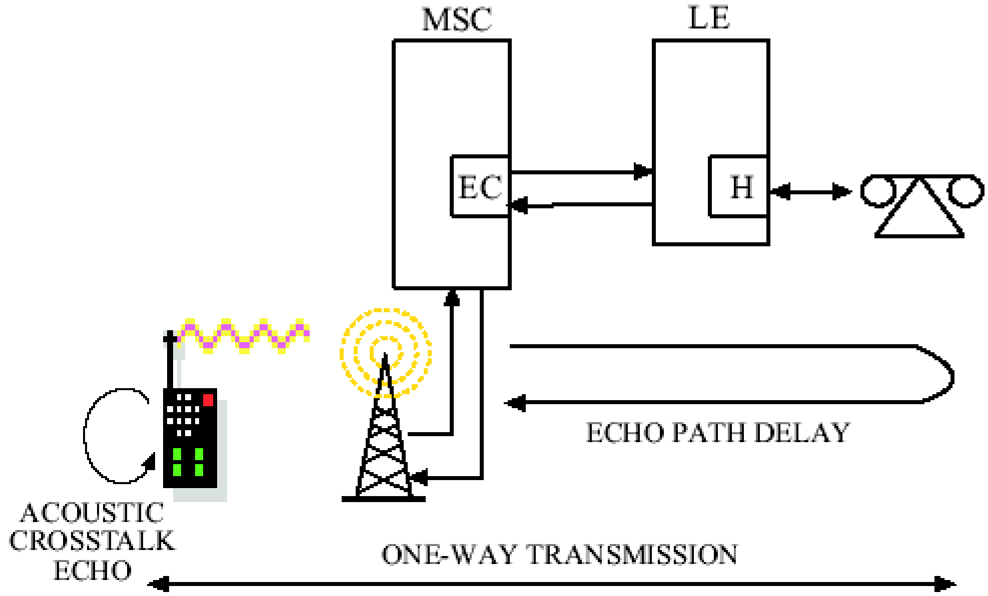

Figure 2.6: Cellular to Fixed Telephone Connection

2.6 Teleconference/Videoconference Communication Systems

When the telephone connection is between hands-free telephones or between two conference rooms, then an acoustic echo problem emerges that is due to the reflection of the loudspeaker's sound waves from the boundary surfaces and other objects back to the microphone. This acoustic echo can be removed using an adaptive filter as illustrated in Figure 2.7. The adaptive filter attempts to synthesize a model of the acoustic echo at its output.

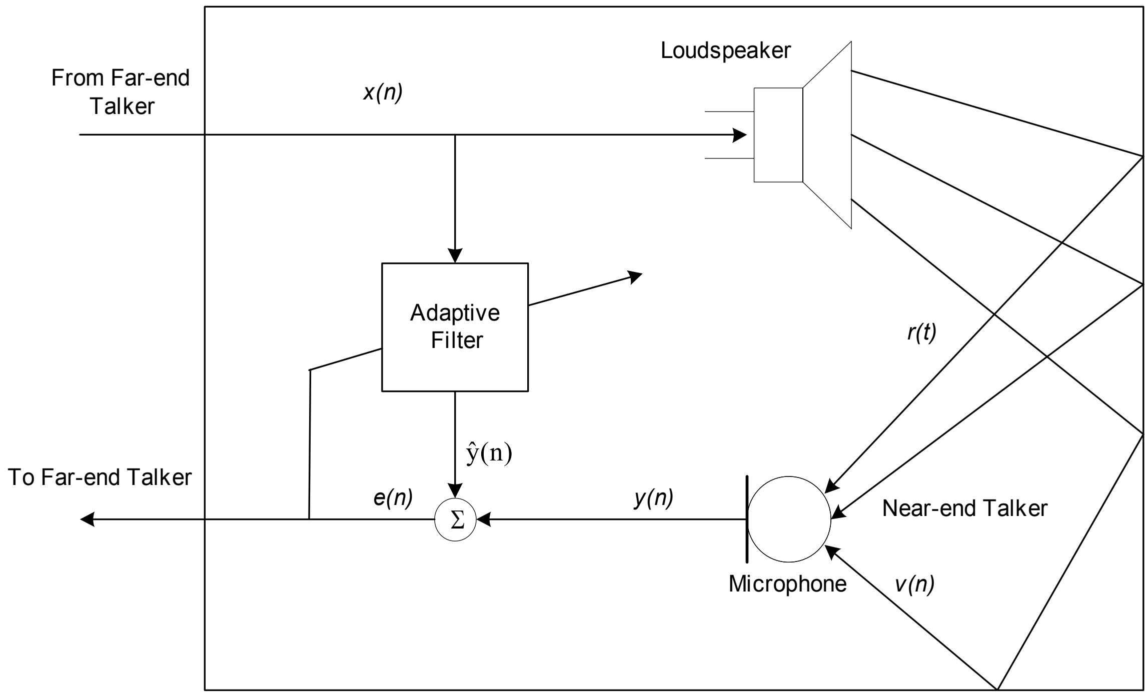

Figure 2.7: Adaptive Acoustic Echo Cancellation in an Enclosed Environment

Adaptive acoustic echo cancellation is a more challenging problem than the network echo cancellation for the following main reasons:

- The impulse response of the acoustic echo path is several times longer, between 100 to 500 msec. than that of the network echo path.
- The characteristics of the acoustic echo path are more non-stationary due to opening and closing of a door or movement of people inside the room while the network echo path is almost stationary.
- The acoustic echo path has a mixture of linear and nonlinear characteristics. The reflection of acoustic signals inside a room is almost linearly distorted. However, the loudspeaker does introduce nonlinearity. The main causes of this nonlinearity are the suspension nonlinearity that affects distortion at low

frequency and the inhomogeneity of flux density that produces nonlinear distortion at large input signal levels.

Due to the above mentioned reasons, the acoustic echo cancellers, (AECs), are required to have more computing power in order to compensate for the longer impulse response and to produce faster converging algorithms [2].

CHAPTER 3

THE ECHO CANCELLATION ALGORITHM

This chapter discusses the echo cancellation algorithm for a VoIP environment. The basic idea behind the algorithm, its terminology, modes of operation and the problems addressed by the algorithm are discussed in detail.

3.1 Basic Echo Canceller

A basic echo canceller used to remove echo in telecommunication networks is presented in Figure 3.1.

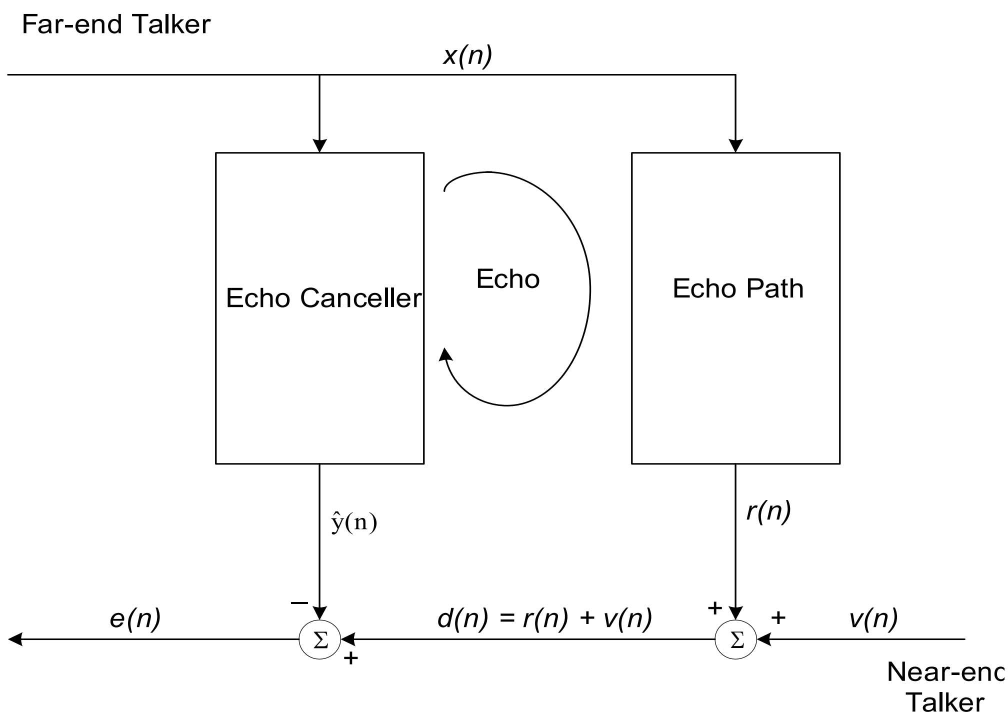

Figure 3.1: A Basic Echo canceller

The echo canceller mimics the transfer function of the echo path in order to synthesize a replica of the echo. Then the echo canceller subtracts the synthesized replica from the combined echo and near-end speech or disturbance signal to obtain the near-end signal. However, the transfer function is unknown in practice. Therefore, it must be identified. This problem can be solved by using an adaptive filter that gradually matches its estimated impulse response, $\hat{h}$, to that of the impulse response of the actual echo path, h. This process is illustrated in Figure 3.2. The echo path is highly variable and can even depend on such things as the movement of people in the room as well as other things. These variations are accounted for by the adaptive control loop, which is built into the canceller.

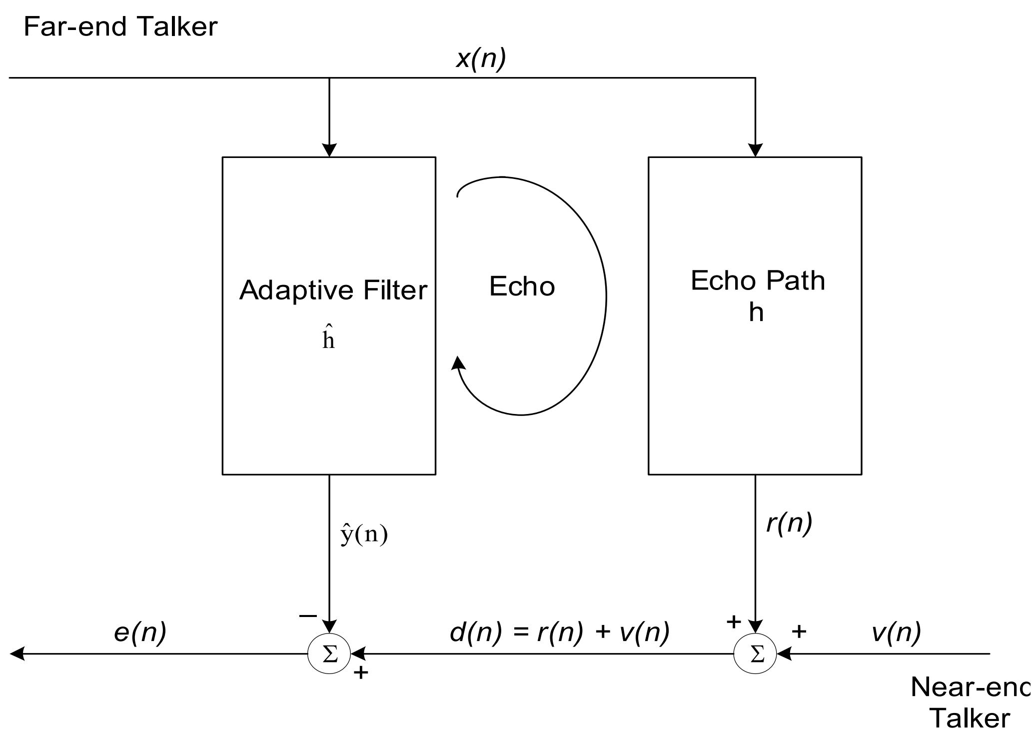

Figure 3.2: A Generic Adaptive Echo Canceller

The estimated echo, $\hat{y}(n)$, is generated by passing the reference input signal, *x(n)*, through the adaptive filter, $\hat{h}(n)$, that will ideally match the transfer function of the echo path, *h(n)*. The echo signal, *r(n)*, is produced when *x(n)* passes through the echo path. The echo *r(n)* plus the near-end talker or disturbance signal, *v(n)*, constitute the desired response,

$$d(n) = r(n)+v(n), \tag{3.1}$$

for the adaptive canceller. The two signals *x(n)* and *r(n)* are correlated since the later is obtained by passing *x(n)* through the echo path. The error signal *e(n)* is given by

$$e(n) = d(n) - \hat{y}(n). \tag{3.2}$$

In the ideal case, *e(n) = v(n),* which represents the case when the adaptive echo canceller is perfect.

Similar to the echo suppressors, adaptive echo cancellers also face the problem of double talking when both near and far end speakers talk simultaneously. If double talk occurs, the system may try to adjust the adaptive filter parameters to imperfectly cancel the near-end talker signal. This will result in making large corrections to the estimated echo path, $\hat{h}$, in an attempt to mimic h. In order to avoid this possibility the coefficients in the adaptive filter must not be updated as soon as double talking is detected as illustrated in Figure 3.3. The design of a good double talking detector is difficult. Even with the assumption of a fast-acting detector, there is still a possibility of changes occurring in the echo channel during the time that the echo canceller is not updated, which leads to increasing amount of uncancelled echoes. Fortunately, the duration of double talking is usually short. In addition to these problems, it sometimes occurs that a well-working echo canceller leaves some residual uncancelled echo. In such a case, a nonlinear processor is used to remove the residual echo. The goal of the nonlinear processor is to block this small unwanted signal if the signal magnitude is lower than a certain small threshold value during single talking. The nonlinear processor will only distort and not block the near-end signal during double talking. The distortion is generally unnoticeable and the processor does not have to be removed during double talking [2].

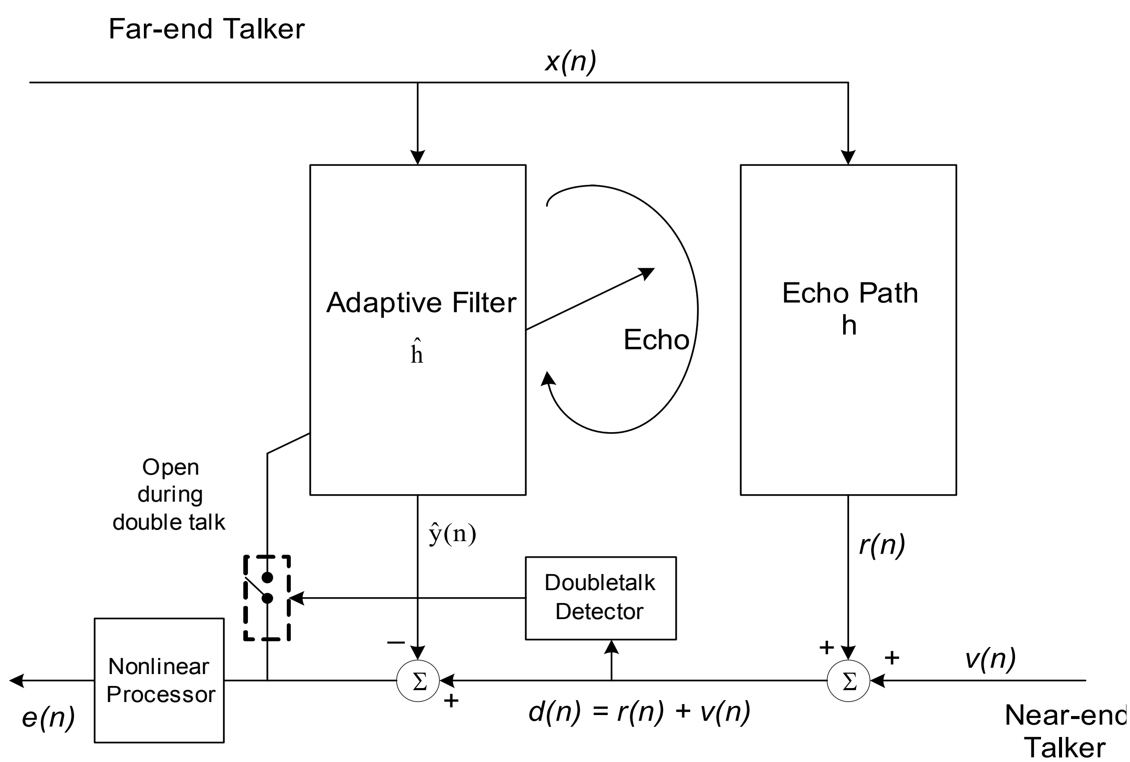

Figure 3.3: Echo Canceller with Doubletalk Detector and Nonlinear Processor

3.2 Components of an Acoustic Echo Canceller (AEC)

The previous section attempted to give some valuable first hand knowledge on the functioning of a basic echo canceller. The following sections offer a detailed theoretical and mathematical account of the three fundamental components of echo cancellers. The three fundamental components that combine to form an echo canceller are:

1. Adaptive Filter
2. Doubletalk Detector
3. Nonlinear Processor

3.3 Adaptive Filtering

As previously demonstrated, the best solution for reducing the echo is to use some form of adaptive algorithm. The theory behind such an algorithm and the reasons for choosing that algorithm will be described in this section. Basically filtering is a signal processing technique whose objective is to process a signal in order to manipulate the information contained in the signal. In other words, a filter is a device that maps its input signal into another output signal by extracting only the desired information contained in the input signal. An adaptive filter is necessary when either the fixed specifications are unknown or time-invariant filters cannot satisfy the specifications. Strictly speaking an adaptive filter is a nonlinear filter since its characteristics are dependent on the input signal and consequently the homogeneity and additivity conditions are not satisfied. Additionally, adaptive filters are time varying since their parameters are continually changing in order to meet a performance requirement. In a sense, an adaptive filter is a filter that performs the approximation step on line.

3.3.1 Least Mean Square (LMS) Algorithm

The least mean square, (LMS), is a search algorithm that is widely used in various applications of adaptive filtering. The main features that attracted the use of the LMS algorithm are low computational complexity, proof of convergence in stationary environments and stable behavior when implemented with finite precision arithmetic. Figure 3.4 illustrates how such an algorithm works. A path that changes the signal *x* is called *h*. Transfer function of this filter is not known in the beginning. The task of the LMS algorithm is to estimate the transfer function of the filter. The result of the signal distortion is calculated by convolution and is denoted by *r*. In this case *r* is the echo and *h* is the transfer function of the hybrid. The near-end speech signal *v* is added to the echo. The adaptive algorithm tries to create a filter *w*. The transfer function of the filter is an estimate of the transfer function for the hybrid. This transfer function in turn is used for calculating an estimate of the echo. The echo estimate is denoted by $\hat{r}$.

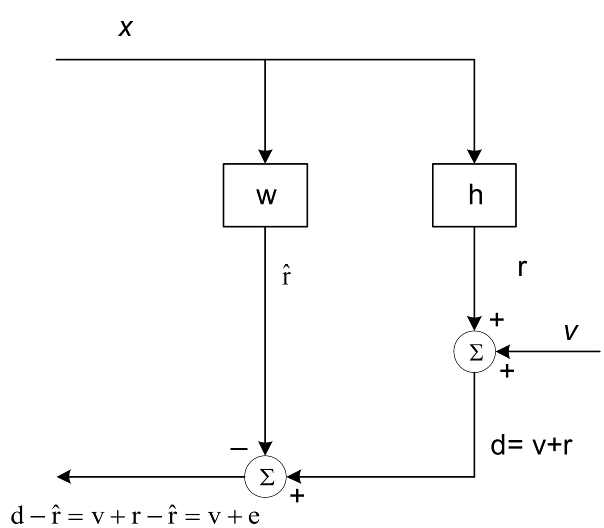

Figure 3.4: LMS Algorithm

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

$$v + r - \hat{r} = v + e, \tag{3.3}$$

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

$$w = b_0 + b_1 z^{-1} + \cdots + b_{L-1} z^{-(L-1)}. \tag{3.4}$$

A perfect FIR filter is linear, time-invariant and stable in a BIBO sense. However, in a real-time environment, linearity is never a possibility and the first criterion is not fulfilled so the filter can never be perfect. Updating of the filter weights is realized in accordance with

$$w(k+1) = w(k) - \mu\, g_w(k) \tag{3.5}$$

for $k = 0,1,2,\cdots$ where $g_w(k)$ represents an estimate of the gradient vector and μ is the convergence factor or step size.

3.3.1.1 Generic LMS Algorithm [3]

The general case of the LMS algorithm is presented in Figure 3.5.

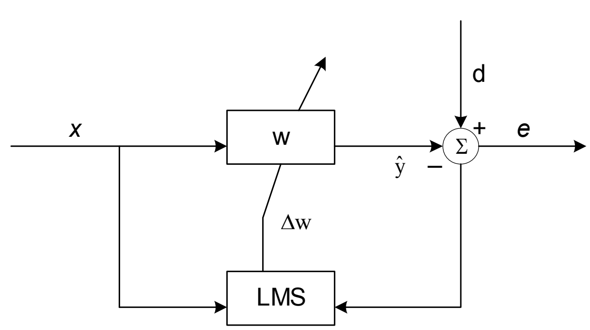

Figure 3.5: Generic LMS Algorithm

Figure 3.5 shows that

$$e(k) = d(k) - \hat{y}(k) = d(k) - x^T(k)\, w(k), \tag{3.6}$$

where $w(k)$ is a vector containing the filter weights $[b_0,\ b_1,\ b_2,\ \cdots,\ b_0]$ and $x(k)$ represents the vector $[x(n),\ x(n-1),\ \cdots,\ x(n-L)]^T$. L is the length of the adaptive filter.

The derivation of the gradient estimate $g_w(k)$ is provided next.

The Wiener solution is given by

$$w_o = R^{-1}\, p \tag{3.7}$$

where

$$R = E\,[\,x(k)\, x^T(k)] \tag{3.8}$$

and

$$p = E[d(k)\, x(k)], \tag{3.9}$$

assuming $d(k)$ and $x(k)$ are jointly wide sense stationary. If good estimates of the matrix R, denoted by $\hat{R}(k)$, and of vector p, denoted by $\hat{p}(k)$, are available, a steepest-descent based algorithm can be used to search the Wiener solution is as follows

$$\begin{aligned} w(k+1) &= w(k) - \mu\, g_w(k) \\ &= w(k) + 2\mu\,(\hat{p}(k) - \hat{R}(k) w(k)). \end{aligned} \tag{3.10}$$

One possible solution is to estimate the gradient vector by employing instantaneous estimates for R and p, which are given by:

$$\hat{R}(k) = x(k)\, x^T(k), \tag{3.11}$$

and

$$\hat{p}(k) = d(k)\, x(k). \tag{3.12}$$

Then the gradient estimate $g_w(k)$ is given by

$$g_w(k) = 2d(k)x(k) + 2x(k)x^T(k)w(k)$$
$$= 2x(k)(d(k) + x^T(k)w(k))$$
$$= 2e(k)x(k). \tag{3.13}$$

The resulting gradient-based algorithm is known. It minimizes the mean of the squared error, as the least-mean square (LMS) algorithm, whose updating equation is given by

$$w(k+1) = w(k) + 2\mu e(k)x(k). \tag{3.14}$$

Table 3.1 presents the steps associated with the LMS algorithm in tabular form.

Table 3.1: LMS Algorithm

Initial Condition	$x(0) = w(0) = [0,\cdots,0]^T$
For each instant of time, k = 1, 2, ···, compute	
Filter output:	$y(k) = x(k)^T w(k)$
Estimation Error:	$e(k) = d(k) - \hat{y}(k)$
Tap-Weight Adaptation:	$w(k+1) = w(k) + 2\mu e(k)x(k)$

3.3.2 Normalized Least Mean Square (NLMS) Algorithm [3]

There are a number of algorithms for adaptive filters, which are derived from the conventional LMS algorithm. The objective of the alternative LMS-based algorithms is either to reduce computational complexity or convergence time. The normalized LMS, (NLMS), algorithm utilizes a variable convergence factor that minimizes the instantaneous error. Such a convergence factor usually reduces the convergence time but increases the misadjustment.

The updating equation of the LMS algorithm can employ a variable convergence factor μ_k in order to improve the convergence rate. In this case, the updating formula is expressed as

$$w(k+1) = w(k) + 2\,\mu_k\, e(k)x(k) = w(k) + \Delta\hat{w}(k)\,, \tag{3.15}$$

where μ_k must be chosen with the objective of achieving a faster convergence. The value of μ_k is given by

$$\mu_k = \frac{1}{2\hat{x}(k)x(k)}\,. \tag{3.16}$$

Using the variable convergence factor the updating equation for the NLMS algorithm is given by

$$w(k+1) = w(k) + \frac{e(k)x(k)}{x^T(k)x(k)}\,. \tag{3.17}$$

Usually a fixed convergence factor μ_n is introduced in the updating formula in order to control the misadjustment since all the derivations are based on instantaneous values of the squared errors and not on the MSE. Also a parameter γ should be included in order to avoid large steps when $x^T(k)x(k)$ becomes small. Then the coefficient updating is by

$$w(k+1) = w(k) + \frac{2\mu_n}{\gamma + x^T(k)x(k)}e(k)x(k) \tag{3.18}$$

Table 3.2 presents the steps associated with the NLMS algorithm in tabular form.

Table 3.2: NLMS Algorithm

Initial Condition	$0 < \mu_n \leq 2$ $x(0) = w(0) = [0,\cdots,0]^T$ γ = a small constant
For each instant of time, k = 1, 2, ⋯, compute	
Filter output:	$y(k) = x(k)^T w(k)$
Estimation Error:	$e(k) = d(k) - \hat{y}(k)$
Tap-Weight Adaptation:	$w(k+1) = w(k) + \dfrac{2\mu_n}{\gamma + x^T(k)x(k)} e(k)x(k)$

3.4 Double Talk Detector (DTD)

An important characteristic of a good echo canceller is its performance during double talk. The condition where both ends, the near-end and the far-end, are speaking is referred to as double talk. If the echo canceller does not detect a double talk condition properly the near end speech will cause the adaptive filter to diverge. Therefore, it is important to have a reliable double-talk detector.

A DTD is used with an echo canceller to sense when the far-end speech is corrupted by the near-end speech. The role of this important function is to freeze adaptation of the model filter, $\hat{h}$, when the near-end speech, *v*, is present in order to avoid divergence of the adaptive algorithm. The far-end talker signal, *x*, is filtered with the impulse response, *h*, and the resulting signal. The echo is added to the near-end speech signal, *v*, in order to obtain the corrupted signal

$$d(n) = \mathrm{H}^T x(n) + v(n) \tag{3.19}$$

where

$$\mathrm{H} = [\mathrm{H}_0, \mathrm{H}_1, \cdots, \mathrm{H}_{L-1}]^{\mathrm{T}} \tag{3.20}$$

and

$$x(n) = [x(n), x(n-1), \cdots, \cdot x(n-L+1)]^T. \tag{3.21}$$

L is the length of the echo path. The error signal at time n is defined by

$$e(n) = d(n) - \hat{\mathrm{H}}^{\mathrm{T}} x(n). \tag{3.22}$$

This error signal is used in the adaptive algorithm to adjust the L taps of the filter, $\hat{\mathrm{h}}$. For simplicity it is assumed that the length of the signal vector, x, is the same as the effective length of the echo path, h. When v is not present, with any adaptive algorithm, $\hat{\mathrm{h}}$ will quickly converge to an estimate of h, which is the best way to cancel the echo. When x is not present, or very small, adaptation is halted by the nature of the adaptive algorithm. When both x and v are present the near-end talker signal could disrupt the adaptation of $\hat{\mathrm{h}}$ and cause divergence. Therefore, the goal of a double talk detection algorithm is to stop the adaptation of $\hat{\mathrm{h}}$ when the level of v becomes significant in relation to the level of x and to keep the adaptation going when the level of v is negligible [4].

The basic double talk detection process starts with computing a detection statistic and comparing it with a preset threshold. Different methods have been proposed to form the detection statistic. The Geigel algorithm has proven successful in line echo cancellers. However, it does not always provide reliable performance when used in AEC's. Cross-correlation based methods appear to be more suitable for AEC applications. However, for the DTD algorithms only heuristic methods have been used to select the threshold T with little justification for the choice. In addition, there has not been an objective way to evaluate and compare these methods.

3.4.1 The Generic Doubletalk Detection Schemes

Almost all types of doubletalk detectors operate in the same manner. Therefore, the general procedure for handling double talk is described by the following four steps.

1. A detection statistic, ξ, is formed using available signals such as *x*, *d* and *e* and the estimated filter coefficients, $\hat{h}$.
2. The detection statistic, ξ, is compared to a preset threshold, T, (a constant), and double talk is declared if $\xi < T$.
3. Once doubletalk is declared the detection is held for a minimum period of time T_{hold}. While the detection is held the filter adaptation is disabled.
4. If $\xi \geq T$ consecutively over a time T_{hold} the filter resumes adaptation while the comparison of ξ to T continues until $\xi < T$ again.

The hold time, T_{hold}, in steps 3 and 4 is essential to suppress detection dropouts due to the noisy behavior of the detection statistic. Although there are some possible variations most of the DTD algorithms keep this basic form and only differ in how they form the detection statistic.

An optimum decision variable, ξ, for double talk detection should behave as follows:

- if $v = 0$ (doubletalk is not present), $\xi \geq T$
- if $v \neq 0$ (doubletalk is present), $\xi < T$

The threshold T must be a constant, independent of data. Moreover ξ must be insensitive to echo path variations when $v = 0$ [5].

In the following sections discussions of different DTD algorithms such as the Geigel Algorithm, the Cross- correlation Method and the Normalized Cross-Correlation Method are presented. The DTD algorithm used in this research was the Normalized Cross-Correlation Method.

3.4.2 The Geigel Algorithm

One simple algorithm due to A. A. Giegel declares the presence of near-end speech whenever

$$\xi = \frac{\max\{|x(k)|,\cdots,|x(k-N+1)|\}}{|d(k)|} < T \tag{3.23}$$

where N and T are suitably chosen constants. This detection scheme is based on a waveform level comparison between the microphone signal, *d*, and the far-end speech, *x*, assuming the near-end speech, *v*, in the microphone signal will be stronger than the echo. The maximum, or norm, of the N most recent samples of *x* is chosen for the comparison due to uncertain delay in the echo path. The threshold, T, is used to compensate for the energy level of the echo path response, *h*, and is often set to ½ for line echo cancellers since the hybrid loss is typically approximately 6dB. However, for an AEC, it is not easy to set a universal threshold that will work reliably in all the various situations since the loss through the acoustic echo path can vary greatly depending on many factors. For N, one easy choice is to set it equal to the adaptive filter length L [5].

3.4.3 The Cross Correlation Method

This method uses the cross-correlation coefficient vector between x and *d* as a means for double talk detection. The cross-correlation coefficient vector between x and *d* is defined by

$$c_{xd} = \frac{E\{x(n)d(n)\}}{\sqrt{E\{x^2(n)\}E\{d^2(n)}} \tag{3.24}$$

$$= \frac{r_{xd}}{\sigma_x \sigma_d} \tag{3.25}$$

$$= [c_{xd,0} \quad c_{xd,1} \ \ldots\ldots c_{xd,L-1}]^T \tag{3.26}$$

where E denotes the mathematical expectation and $c_{xd,I}$ is the cross-correlation coefficient between $x(n - I)$ and $d(n)$. The idea is to compare

$$\xi = \|c_{xd}\| \tag{3.27}$$

$$= \max |c_{xd,i}|,\ i = 0,1,\cdots,L-1 \tag{3.28}$$

to a threshold level T. The decision rule is then very simple. If $\xi \geq$ T, double talk is not present and if $\xi <$ T, double talk is present.

The fundamental problem with this method is that the cross-correlation coefficient vectors are not well normalized. In general, it is assumed that $\xi \leq 1$. Therefore, if v = 0, it does not mean that $\xi = 1$ or any other known value. The value of ξ is not known in general. The amount of correlation will depend greatly on the statistics of the signal and of the echo path. As a result, the best value of T will vary from one experiment to another. There is no natural threshold level associated with the variable ξ when v= 0. These complexities lead to another DTD algorithm, which is termed the Normalized Cross-Correlation method. This method is simply a modification of the existing Cross-Correlation Method [4].

3.4.4 Normalized Cross Correlation Method

In this method a new normalized cross-correlation vector between a vector x and a scalar d is derived. Suppose that $v = 0$. In this case

$$\begin{aligned} R_{dd} &= E\{d(n)d^T(n)\} \\ &= H^T R_{xx} H \end{aligned} \quad (3.29)$$

where

$$R_{xx} = E\{x(n\ x^T(n)\}. \quad (3.31)$$

Since

$$d(n) = H^T x(n), \quad (3.32)$$

$$R_{xd} = R_{xx} H, \quad (3.33)$$

which allows R_{dd} to be rewritten as

$$R_{dd} = R^T{}_{xd} R^{-1}{}_{xx} R_{xd}. \quad (3.34)$$

In general, for $v \neq 0$,

$$R_{dd} = R^T{}_{xd} R^{-1}{}_{xx} R_{xd} + R_{vv} \quad (3.35)$$

where

$$R_{vv} = E\{v(n)v^T(n)\} \quad (3.36)$$

is the covariance matrix of the near-end speech. The new decision variable is obtained by dividing equation(3.35) by R_{dd} and extracting the square root, which yields

$$\xi = \sqrt{R^{T}{}_{xd}\ R^{-1}{}_{xx} R_{xd} R^{-1}{}_{dd}} \tag{3.37}$$

$$= \left\| c_{xd} \right\| \tag{3.38}$$

where

$$c_{xd} = R^{-1/2}{}_{xx} R_{xd} R^{-1/2}{}_{dd} \tag{3.39}$$

is the normalized cross-correlation vector between *x* and *d*. Substituting equation (3.33) and equation (3.35) into equation (3.37) produces the decision variable, which is given by

$$\xi = \frac{\sqrt{H^{T} R_{xx} H}}{\sqrt{H^{T} R_{xx} H + \sigma^{2}{}_{v}}}\ . \tag{3.40}$$

Equation (3.30) shows that for $v = 0$; $\xi = 1$ and for $v \neq 0$; $\xi < 1$. Therefore, the threshold value can be set tone (1). It should also be noted that ξ is not sensitive to changes of the echo path when $v = 0$ [4], [5].

3.5 Nonlinear Processor (NLP)

A nonlinear processor, (NLP), is a signal processing circuit or algorithm that is placed in the speech path after echo cancellation in order to provide further attenuation or removal of residual echo signals that cannot be removed completely by an echo canceller. A non-linearity, a distortion, or an added noise signal are examples of signals that cannot be fully cancelled by an echo canceller. Therefore, these signals are typically removed or attenuated by a nonlinear processor.

3.5.1 Noise Gate as a NLP

In this research a noise gate was used as a NLP, which is a type of dynamic processor. Noise gates belong to the family of expanders. As the name implies, it increases the dynamic range of a signal such that low-level signals are attenuated while the higher-level portions are neither attenuated nor amplified. The noise gate expansion can be taken to the extreme where it will heavily attenuate the input or eliminate it entirely leaving only silence.

While expanders are quite difficult to use effectively, noise gates are a very common and effective way of reducing the apparent noise level in audio signals. The noise gate offers a method of turning down the gain of an audio signal when the signal level drops below some threshold value. The threshold value needs to be high enough that only the background noise falls below but not so high that the audio signals are cut off prematurely. Noise gates are most often used to eliminate noise or hiss that may otherwise be amplified.

3.5.2 A Generic Expander

Figure 3.6 presents the basic structure of an expander.

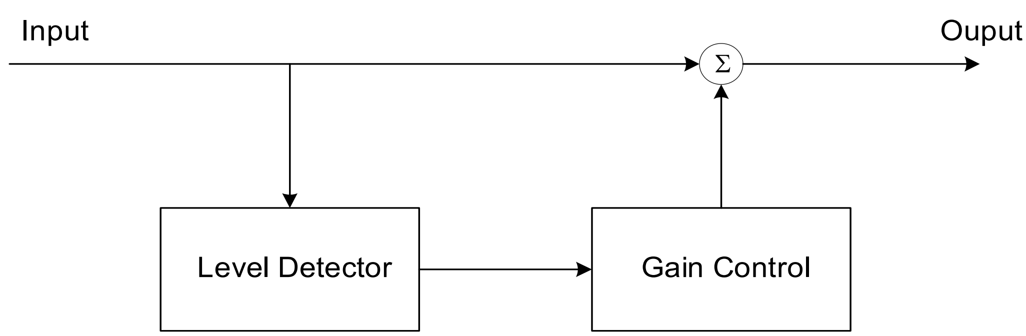

Figure 3.6: Basic Block Diagram of an Expander

An expander is essentially an amplifier with a variable gain control. The level of the input signal is sensed by the level detector and applied to the gain control element. The gain is never greater than one and is controlled by the level of the input signal. When the input signal level is higher than a threshold value the expander has a unity gain and acts as a normal unity gain amplifier. When the input signal level drops below the threshold the gain decreases, which makes the signal even lower or the signal is completely removed depending on the threshold value. This feature drove the choice of using a noise gate as the NLP since the signal level of the echo is very much less than that of the near-end signal.

The input/output relationship of the expander is represented in a simple graph, which is presented in Figure 3.7. The level of the input signal is given by the horizontal axis and the output level is given by the vertical axis. When the slope of the line is unity, angled at 45 degrees, the gain of the expander is one (1). Therefore, the output level is identical to the input level. A change in the line's slope means a change in the expander's gain. For the expander, part of the line will have a larger slope. The point where the slope of the line changes is called the threshold, which is adjustable in many expanders. When the input signal level is above the threshold nothing happens. However, when the input signal level drops below the threshold the gain reduction starts. The gain reduction lowers the input level by increasing or expanding the dynamic range.

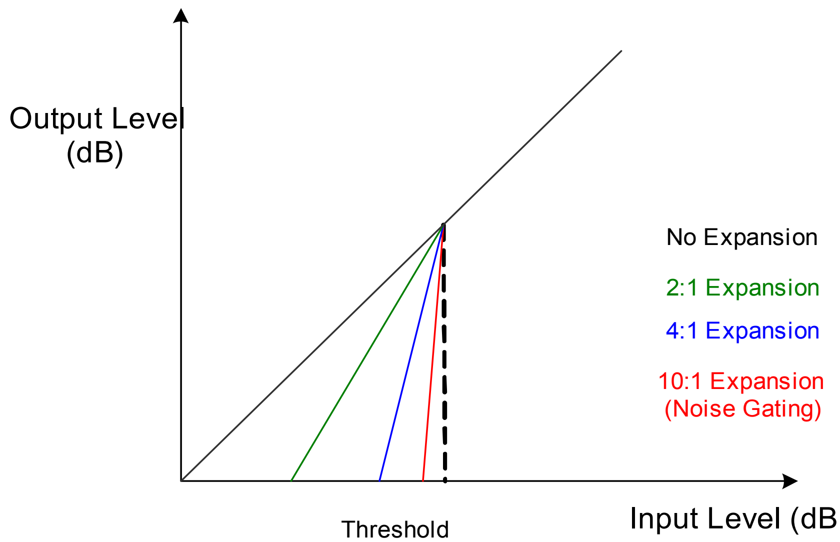

Figure 3.7: Input / Output Characteristics of an Expander

The amount of expansion that is applied is usually expressed as a ratio such as 2:1 or 4:1. This implies that while the input is below the threshold a change in the input level produces a change in the output that is two times or four times as large. Therefore, with a 4:1 expansion ratio and the input level below the threshold a dip of 3 dB in the input will produce a drop of 12 dB in the output [6].

3.5.3 Noise Gate

When an expander is used with extreme settings where the input/output characteristic becomes almost vertical below the threshold and when the expansion ratio larger than 10:1, the expander is often termed a noise gate. In this case, the input signal may be very heavily attenuated or removed entirely. Therefore, the expander acts like an on/off switch for signals. When the signal is high enough, the switch is on and the input appears at the output. However, when the signal drops below the threshold the switch is off and there is no output. Hence, when the near-end signal passes through this on/off switch or noise gate, because of the high signal level the switch is on and attenuation does not occur. However, when the echo signal passes the switch is off and the echo is completely removed or highly attenuated depending on the threshold. Hence the important aspect of this device is the choice of a correct threshold value.

Since the level sensing function is a short time average it takes some time for a change in the input level to be detected, which triggers a change in the gain. In general an expander is characterized by its attack and release times. The attack time is the time required for the expander to restore the gain to one once the input level rises above the threshold. Likewise, the time taken for the expander to reduce its gain after the input drops below the threshold is the release time. The attack and release times give the expander a smoother change in the gain rather than abrupt changes that may produce pops and/or other noise. Figure 3.8 illustrates how the attack and release times affect an example input signal [7].

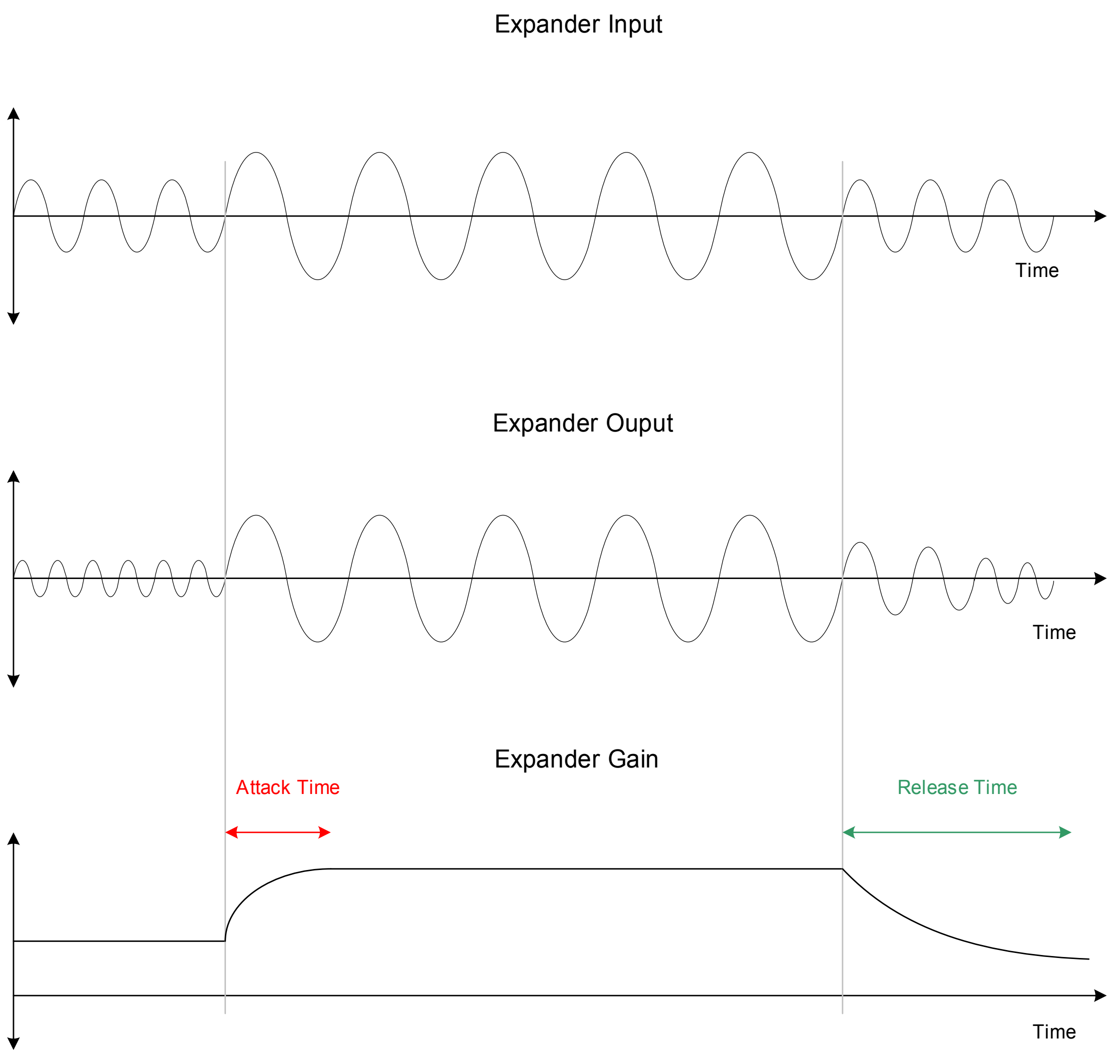

Figure 3.8: The Effect of an Expander on a Signal

Only the middle portion of the input is above the expander's threshold value. However, it takes some time for the expander to increase the gain when the input level rises above the threshold. When the input level drops below the threshold the expander gradually reduces its gain. Therefore, a noise gate fulfilled this research's need for a NLP. Another important aspect of the selection was that the noise gate does not facilitate clipping of talker's signal, which is very common in the with other NLP types.

CHAPTER 4

SIMULATION AND RESULTS

The previous chapters provide a detailed sketch of an Acoustic Echo canceller, (AEC). In this chapter the flowchart for the software simulation and the results of simulation of the AEC algorithm, which was performed in MATLAB are discussed. The idea that drove the simulation was to show that convincible results could be achieved in the software environment.

4.1 Why MATLAB?

MATLAB is a powerful, general-purpose, mathematical software package. MATLAB possesses excellent graphics and matrix handling capabilities. It integrates mathematical computing in a powerful language to provide a flexible environment for technical computing. The salient features of MATLAB are its in-built mathematical toolboxes and graphic functions. Additionally, external routines that are written in other languages such as C, C++, Fortran and Java, can be integrated with MATLAB applications. MATLAB also supports importing data from files and other external devices. Most of the functions in MATLAB are matrix-oriented and can act on arrays of any appropriate dimension. MATLAB also has a separate toolbox for signal processing applications, which provided simpler solutions for many of the problems encountered in this research.

The MATLAB software environment suited the needs of this research for the following reasons:

- The input signals (far-end and near-end talker signals) were voices. These voices were stored as *wav* files and the *wav* files were easily imported into the code.
- The intermediate signals (echo signals) and output signals (error signal and signals obtained after echo cancellation) were obtained as *wav* files. Thus the audio of the voice signals could be literally be heard, which aided immensely judgments with respect to the results obtained.
- The signal processing toolbox has in-built functions for almost all signal processing applications. The toolbox helped the efficiency of the code since these functions could be called wherever necessary instead of writing separate sub-routines.
- Since MATLAB supports graphics, the results of a simulation could be presented in a graphical format with ease.

4.2 Simulation Flowchart

The flowchart for the simulation of the echo canceller algorithm is presented in Figure 4.1.

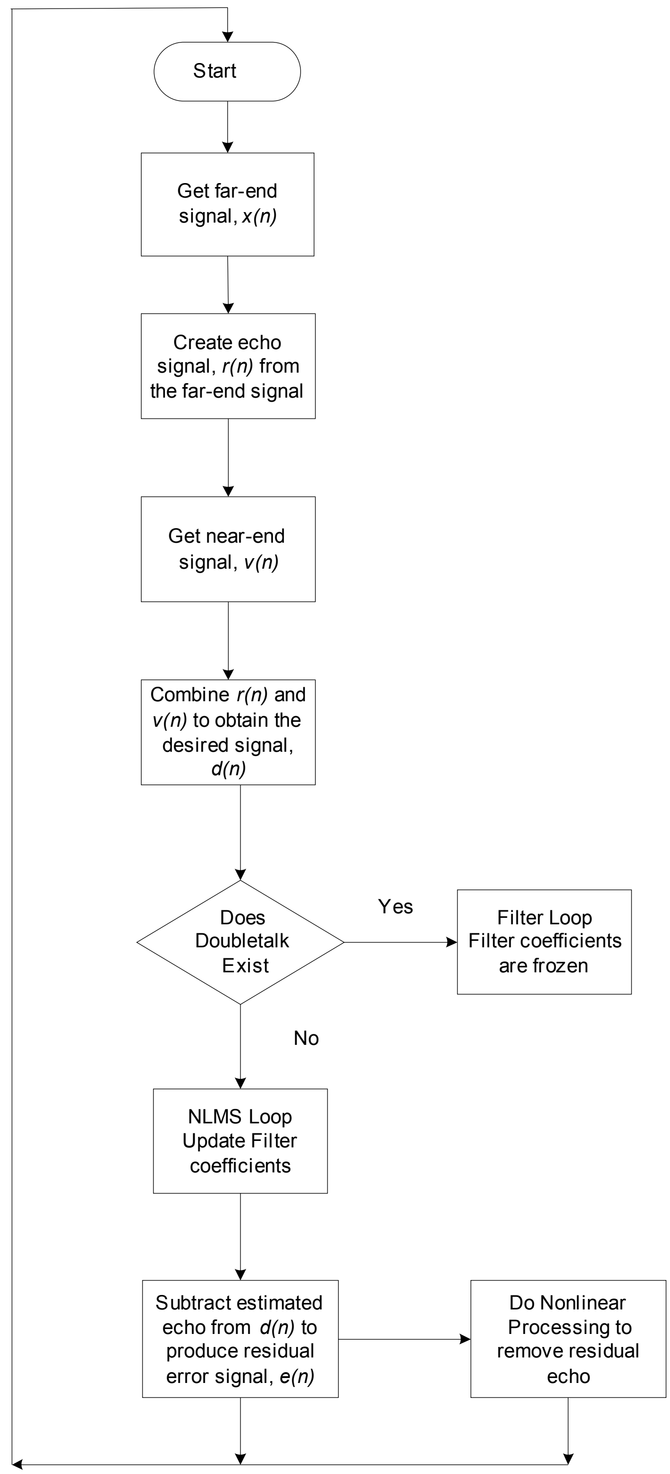

Figure 4.1: Flowchart of the MATLAB Simulation

4.3 Description of the Simulation Setup

This section describes the simulation environment, its requirements and the procedures adopted.

1. The input signals, both far-end and near-end signals, were simulated and given to the AEC, which executed on a PC with the MATLAB environment.
2. The input signals seven seconds in duration.
3. A sampling rate of 8000 Hz was used for all the signals in the simulation.
4. The graphs plotted have x-axes denoting the time and y-axes denoting the amplitude or magnitude of the signal.

4.4 Results

This section presents a graphical representation of the results obtained by simulating the algorithm in MATLAB. The plot of the far-end signal $x(n)$ is presented in Figure 4.2.

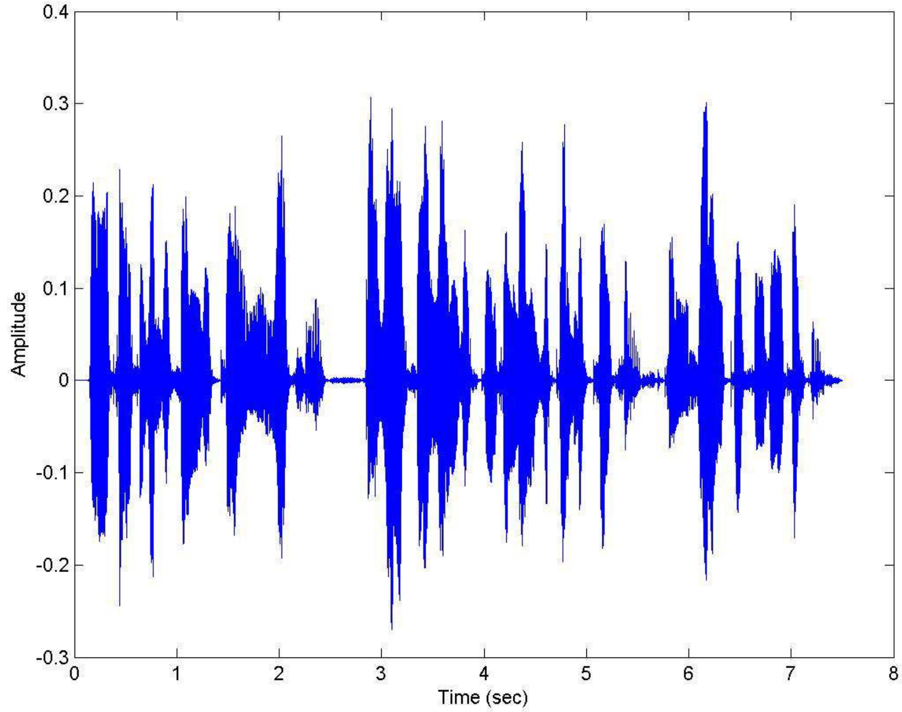

Figure 4.2: Plot of the Far-end Signal, $x(n)$

The far-end signal was delayed and scaled in order to produce the echo signal, $r(n)$, which is presented in Figure 4.3. The echo signal was produced when the far-end signal, $x(n)$, passed through the echo path, h.

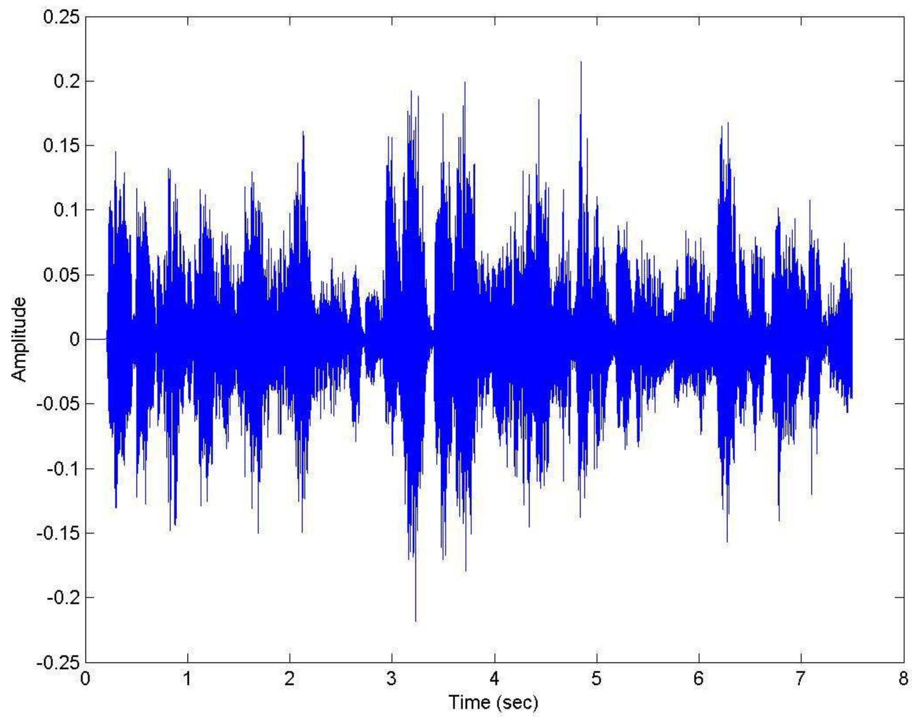

Figure 4.3: Plot of the Echo Signal, *r(n)*

The echo signal was added to the near-end signal, *v(n)*, in order to produce the desired signal, *d(n)*, which became the input for the adaptive filter. The plot of the near-end signal, *v(n)*, is presented in Figure 4.4 and the plot of the desired signal, *d(n)*. is presented in Figure 4.5.

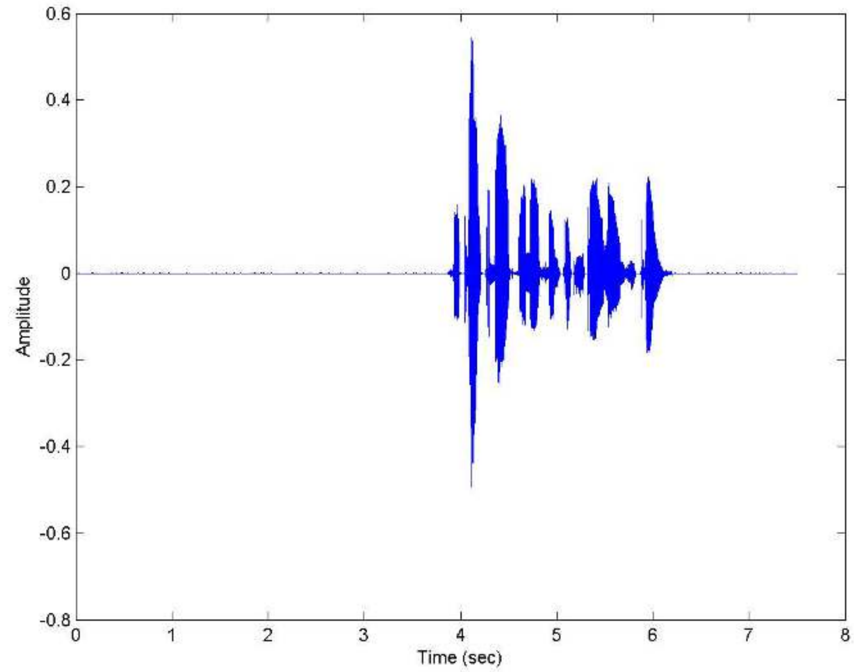

Figure 4.4: Plot of the Near-end Signal, *v(n)*

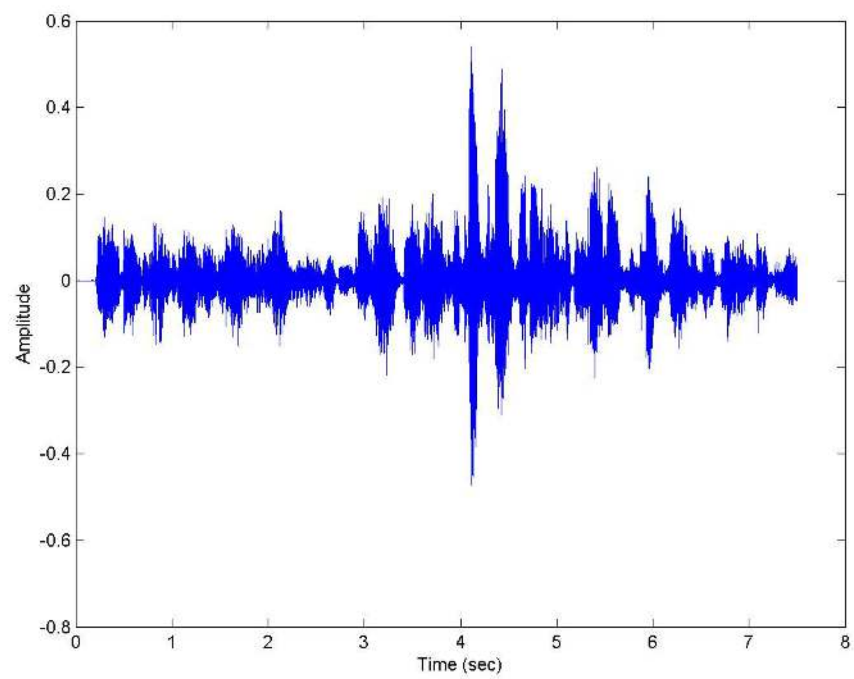

Figure 4.5: Plot of the Desired Signal, *d(n)*

The desired signal, *d(n)*,was passed through the adaptive filter and the double talk detector. For the purpose of adaptive filtering the NLMS algorithm was used during the simulation. The algorithm used the normalized cross correlation algorithm for double talk detection.

Various parameters for the NLMS algorithm such as the convergence factor, μ_n, and γ had to be set in order to avoid misadjustment. Additionally, the length of the filter had to be established beforehand. The values of these parameters, which were used in the simulation, are

- Length of the filter, N = 512
- Convergence factor, $\mu_n = 1.9$. This value was found to produce faster convergence of the NLMS algorithm.
- A small constant, $\gamma = 0.9$

For the purpose of the open simulation environment and faster convergence of the algorithm, it was assumed that double talk did not take place during this simulation.

The output of this module is the error signal, *e(n)*, which is presented in Figure 4.6. In the case of an ideal echo canceller the error signal should be the same as that of the near-end signal, *v(n)*. However, due to the presence of residual echo and nonlinearities the error signal, *e(n)*, was not a perfect copy of the near-end signal, *v(n)*.

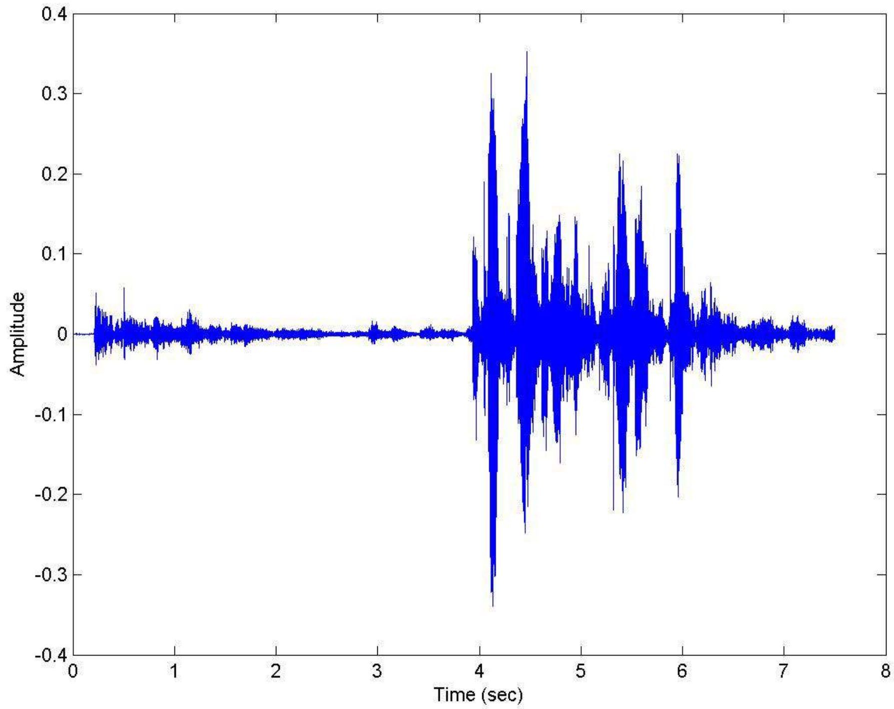

Figure 4.6: Plot of the Error Signal e*(n)*

Since the error signal, *e(n)*, contained a residual echo it was passed through a NLP. As explained earlier, a noise gate was used for the NLP in this research. The purpose of this device was to attenuate the residual echo and to pass on the speech signal without any clipping. Figure 4.7 presents the plot of the error signal after nonlinear processing.

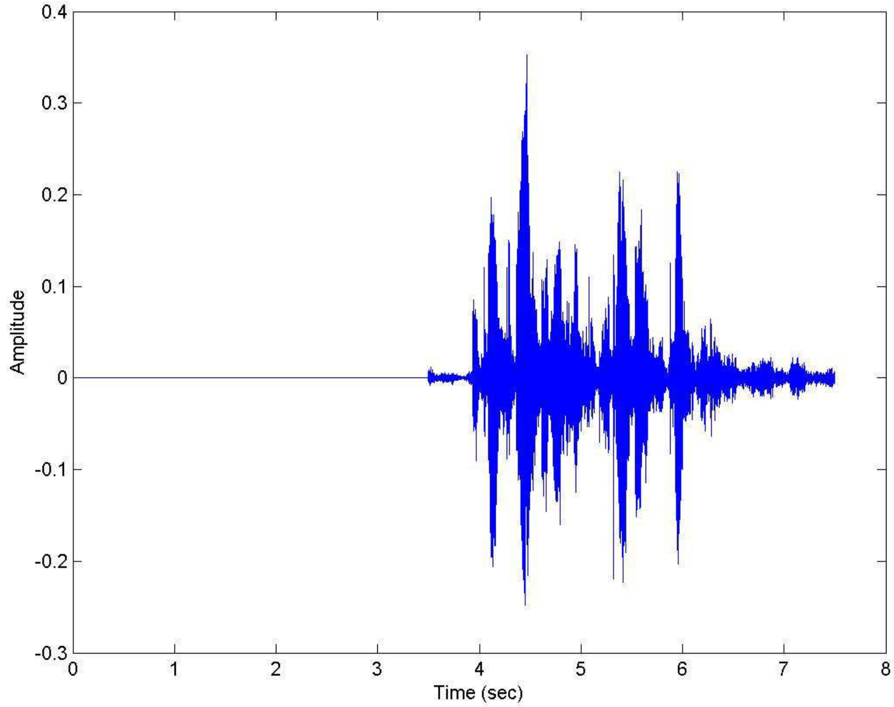

Figure 4.7: Plot of the Error Signal after Nonlinear Processing

Figure 4.7 clearly shows that the residual echo was completely removed and that s no clipping occurred. Therefore, the signal output of the echo canceller was devoid of any significant echoes.

4.5 Evaluation of the Echo Cancellation Algorithm

In order to evaluate the effective working of the algorithm, some basic tests were conducted. This section provides a brief account of these tests.

4.5.1 Convergence Test

The first and paramount test of the algorithm was whether or not the algorithm converged. If the filter coefficients used in the adaptive algorithm did not converge, the code would be useless. Therefore, several tests were performed on the simulated data in order to verify the convergence of the filter coefficients. These tests were conducted by varying the convergence factor, $\mu_{n,}$ and examining the effect on the filter coefficients and the plot of the error signal, *e(n)*. Through careful observation it was determined that a value of 1.9 produced faster convergence.

4.5.2 Echo Return Loss Enhancement (ERLE)

In order to evaluate the quality of the echo cancellation algorithm the measure of ERLE was used. ERLE, measured in dB is defined as the ratio of the instantaneous power of the signal, *d(n)*, and the instantaneous power of the residual error signal, *e(n)*, immediately after cancellation. ERLE measures the amount of loss introduced by the adaptive filter alone. Mathematically it can be expressed as

$$\mathrm{ERLE} = 10\log\frac{P_d(n)}{P_e(n)} = 10\log\frac{E[d^2(n)]}{E[e^2(n)]}. \tag{4.1}$$

For a good echo canceller circuit, an ERLE in the range of 30 dB – 40dB is considered to be ideal. Figure 4.8 presents a plot of the ERLE with the ERLE plotted in dB along the y-axis and the number of samples along the x-axis. The plot of ERLE implies that the ERLE for this algorithm attained the required value.

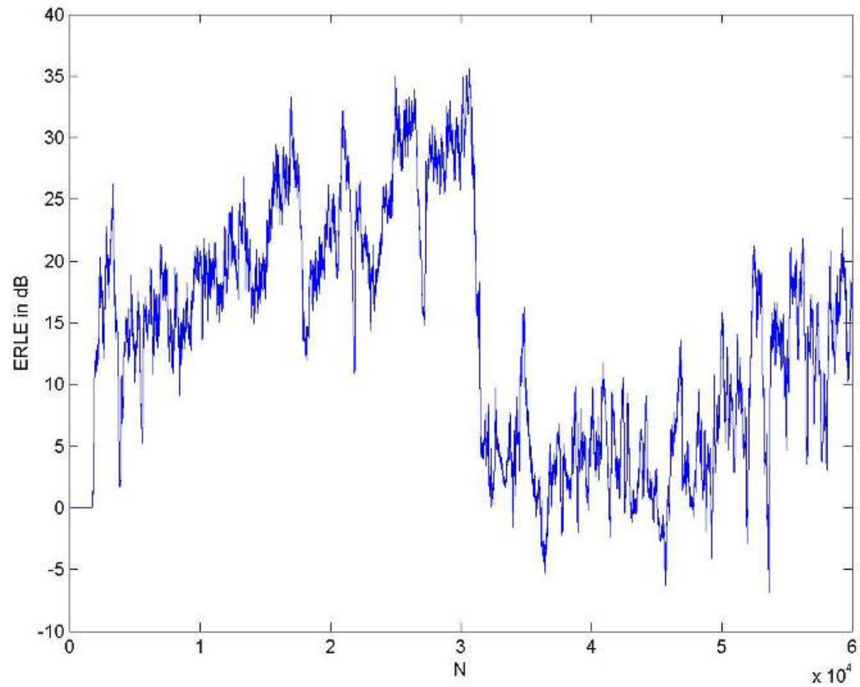

Figure 4.8: Plot of ERLE Vs Number of Samples

4.5.3 Auditory Test

The last test consisted of listening to the output for appropriate cancellation of echoes. The audio of the output signals was presented to a panel of five members with no technical expertise in this field. The panel was almost not able to distinguish the near-end signal, *v(n)*, and the output signal with the residual echo, *e(n)*, removed. Some discrepancies in the audio could be attributed to the fact that the real-time applications cannot escape the factor called noise.

CHAPTER 5

CONCLUTION AND FURTHER WORK

5.1 Conclusions

With the world shrinking into a global village because of superior communications, telephones, both conventional and hands-free sets, occupy a prominent position in solving people's communication needs. One of the major problems in a telecommunication application over a telephone system is echo. The Echo cancellation algorithm presented in this thesis successfully attempted to find a software solution for the problem of echoes in the telecommunications environment. The proposed algorithm was completely a software approach without utilizing any DSP hardware components. The algorithm was capable of running in any PC with MATLAB software installed. Additionally, a new method, which utilized the noise gate device for nonlinear processing was proposed. This new technique is faster and provides almost perfect results for canceling residual echoes without clipping of the reference speech signals. In addition, the results obtained were convincing. The audio of the output speech signals were highly satisfactory and validated the goals of this research.

5.2 Further Work

The algorithm proposed in this thesis presents a solution for single channel acoustic echoes. However, most often in real life situations, multichannel sound is the norm for telecommunication. For example, when there is a group of people in a teleconference environment and everybody is busy talking, laughing or just communicating with each other multichannel sound abounds. Since there is just a single microphone the other end will hear just a highly incoherent monographic sound. In order to handle such situations in a better way the echo cancellation algorithm developed during this research should be extended for the multichannel case.

REFERENCES

[1] Sadaoki Furui and M. Mohan Sondhi, "Advances in Speech Signal Processing", Marcel Dekker, Inc, 1992

[2] Lester S.H Ngia, "System Modeling using Basis Functions and Application to Echo Cancellation", Ph. D. Dissertation, Chalmers University of Technology

[3] Paulo S.R. Diniz, "Adaptive Filtering Algorithms and Practical Implementation", Kluwer Academic Publishers, 1997

[4] J. Benesty, D.R. Morgan and J.H. Cho, "A New Class of Doubletalk Detectors Based on Cross-correlation", IEEE Trans. Speech Audio Processing, vol. 8, pp. 168-172, March 2000

[5] J. Benesty, T. Gansler, D.R. Morgan, M.M. Sondhi and S.L. Gay, "Advances in Network and Acoustic Echo Cancellation", Springer-Verlag, 2001

[6] Jerry C. Whitaker, "Master Handbook of Audio Production", McGraw-Hill, 2003

[7] Eargle, John, "Handbook of Recording Engineering", Van Nostrand Reinhold, 1996

[8] Mu Tian, P.K. Gupta, Marcus Harte and Danny Yip, "Improved Echo Canceller and Implementation", Dialogic Corporation

[9] J.G. Proakis and D.G. Manolakis, "Digital Signal Processing - Principles, Algorithms and Applications", Prentice Hall, 1996

[10] A.V. Oppenheim and R.W. Schafer, "Discrete Time Signal Processing", Prentice Hall, 1996

[11] Sophocles J. Orfanidis, "Optimum Signal Processing - An Introduction", McGraw Hill, 1998

[12] S.Haykin, "Adaptive Filter Theory", Prentice Hall, 1996

[13] Andrzej Borys, "Nonlinear Aspects of Telecommunications", CRC Press, 2001