# IMPLEMENTATION AND COMPARATIVE EVALUATION OF SPEECH COMPRESSION TECHNIQUES

## A DISSERTATION

***Submitted in partial fulfillment of the requirements for the award of the degree of***

**MASTER OF TECHNOLOGY**

**in**

**ELECTRICAL ENGINEERING**

**(With Specialization in Measurement and Instrumentation)**

By

**NAGESWARA REDDY PERAM**

**DEPARTMENT OF ELECTRICAL ENGINEERING**
**INDIAN INSTITUTE OF TECHNOLOGY ROORKEE**
**ROORKEE - 247 667 (INDIA)**
**JUNE, 2007**

I.D. No.-M.T. 394/2007-27/KSA

## CANDIDATE'S DECLARATION

I hereby declare that the work that is being presented in this dissertation report entitled **"IMPLEMENTATION AND COMPARATIVE EVALUATION OF SPEECH COMPRESSION TECHNIQUES"** submitted in partial fulfillment of the requirements for the award of the degree of **Master of Technology** with specialization in **Measurement and Instrumentation,** to the **Department of Electrical Engineering, Indian Institute of Technology, Roorkee**, is an authentic record of my own work carried out, under the guidance of **Dr. R.S Anand**, Associate Professor, Department of Electrical Engineering, Indian Institute of Technology, Roorkee.

The matter embodied in this dissertation report has not been submitted by me for the award of any other degree or diploma.

Date: 29/06/07

Place: Roorkee

P. Nageswara Reddy

**(NAGESWARA REDDY PERAM)**

## CERTIFICATE

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

Date: 29.06.07

Place: Roorkee

**(Dr. R. S. ANAND)** 29.06.07

Associate Professor,

Department of Electrical Engineering,

IIT Roorkee, Roorkee-247 667. INDIA

# ACKNOWLEDGEMENTS

I wish to express my deep sense of gratitude and sincere thanks to my beloved guide Dr. R.S Anand, Associate Professor, Department of Electrical Engineering, IIT Roorkee, for being helpful and a great source of inspiration. His keen interest and constant encouragement gave me the confidence to complete my work. I wish to extend my sincere thanks for his excellent guidance and suggestions for the successful completion of my work.

My heartfelt gratitude and indebtedness goes to all the teachers of Measurement and Instrumentation group who, with their encouraging and caring words, constructive criticism and suggestions have contributed directly or indirectly in a significant way towards completion of this dissertation.

I convey my deep sense of gratitude to Dr. S.P Gupta, Head of Electrical Engineering Department, for providing all the facilities required for my dissertation work.

Special thanks to my friends whose support and encouragement has been a constant source of guidance to me.

It is difficult for me to express my gratitude to my parents for their affection and encouragement. Last but not the least; I am indebted to all my classmates from Measurement & Instrumentation group for taking interest in discussing my problems and encouraging me.

P. Nageswara Reddy

(NAGESWARA REDDY PERAM)

# ABSTRACT

Speech compression is one area of digital signal processing that can be used to convert human speech into an efficient encoded representation that again can be decoded to produce a close approximation. Our main aim of this work is to compress the recorded speech. Speech coding is a lossy type of coding, which means that the output signal does not sound like the input. Now a days speech compression has many applications in the area of telecommunications such as digital cellular phones, voice mail and internet phones and also in high quality speech storage and message encryption.

In this dissertation work three speech compression techniques has been implemented for the purpose of compressing the recorded speech. These are Adaptive Differential Pulse Code Modulation, Linear Predictive Coding and Wavelet Transform. Out of these three, Wavelet Transform Based Speech Compression is a new technique. Wavelets have been successfully used in image compression applications. But less attention has been paid towards its application in the field of the speech compression. The aim of this dissertation work has been centered around implementation and comparison of speech compression techniques. Our comparative evaluation was based on the following parameters.

1. Compression Ratio
2. Signal to Noise Ratio
3. Peak Signal to Noise Ratio
4. Normalized Root Mean Square Error

In this work it is proved that compression ratio in the case of ADPCM and LPC is not variable where as in the case of wavelet transform based speech compression, compression ratio is variable and quality is also quite good with respect to ADPCM and LPC. It is also concluded that quality decreases by increasing the compression ratio. In the case of ADPCM compression ratio is less but the quality is good. In the case of LPC compression ratio is more but the quality is poor. Quality and compression ratio is moderate in the case of wavelet transform based speech compression.

# Chapter – 1
# INTRODUCTION

## 1.1 Introduction

Speech is one of the most important tools that people use to communicate ideas. Almost everyone uses speech daily, and we are so comfortable with speech that we take the ability to speak for granted. Speech compression is the technology of converting human speech into an efficiently encoded representation that can later be decoded to produce a close approximation of the original signal [1]. Speech compression has been and still is a major issue in the area of digital speech processing. Speech coding is the act of transforming the speech signal to a more compact form, which can then be stored with a considerably smaller memory. The main aim of the speech compression is to encode and decode the speech signal. The motivation behind speech compression is the fact that access to unlimited amount of bandwidth is not possible [1]. Compression reduces the amount of data to be transmitted, thereby more efficiently utilizing the available communication bandwidth. So there is a need to code and compress the speech signals.

## 1.2 History of Speech Compression

The first major development in the history of speech compression is the invention of the vocoder in 1939. Pulse code modulation (PCM) was first documented in detail in Cattermole's classic contribution in 1969. However, in 1967 it was recognized that predictive coding provides advantages over memory less coding techniques, such as PCM. Predictive techniques were analyzed in depth by Markel and Gray in their 1976 classic treatise. This was followed shortly by the often cited reference by Rabiner and Schafer. The wave form coding of speech and video signals was comprehensively documented by Jayant and Noll in their 1984 monograph. Atal and Schroeder invented the code excited linear predictive principle during the 1980s. In 1996 Johnson Ihyeh Agbinya invented the wavelet in the speech processing in his work he proved that wavelets concentrate speech energy into bands which differentiate between voiced or unvoiced speech. Before to Jhonson Ihyeh Agbinya W. Kinsner and A. Langi wavelets was introduced in speech as well as image compression and he proved that its performance in terms of the bit rates and signal quality is comparable with other good techniques such as the discrete cosine transform for images and code excited linear predictive coding for speech, but with much less computational burden. In 2003 Abdul Mawla M. A. Najih, Abdul Rahman bin Ramli, V. Prakash, and Syed A. R applied different wavelets on speech compression.

### 1.3 Mathematical Model of Speech Production System

Before going to mathematical model of speech production system it is necessary to discuss briefly about the physical model of speech production system. Speech comes form our mouth due to air that has been pushed form the lungs through vocal tract. In the speech signal two types of sounds are there one is voiced sounds and another one is unvoiced sounds. Voiced sounds come when ever vocal cards vibrate. The rate at which the vocal cards vibrate determines the pitch of the voice. It differs form person to person and also it depends on the age [1]. In the case of women and young children tend to have high pitch that means fast vibration of vocal cards. Where as in the case of adult males tend to have low pitch that means slow vibration of vocal cards. For unvoiced sounds the vocal cards does not vibrate but that remain constantly opened. Different sounds can be produced by changing the shape of the vocal tract. The amount of air coming form the lungs determines the loudness of the voice signal.

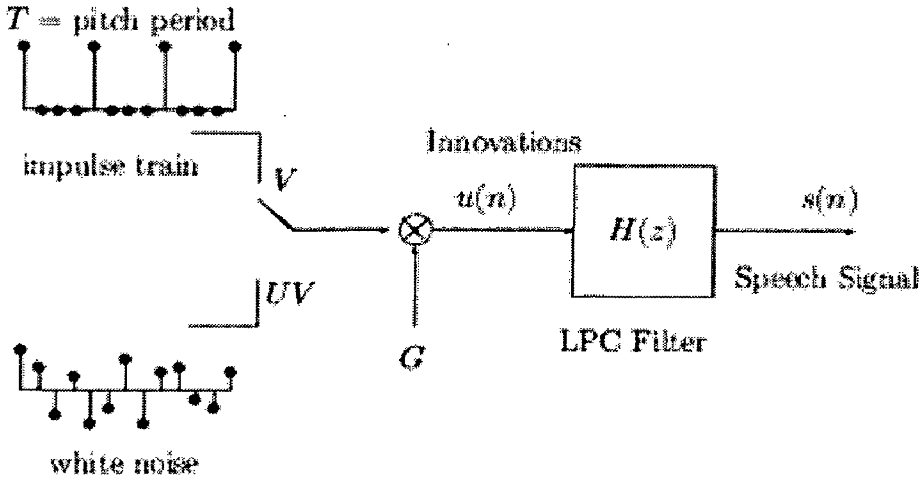


**Figure 1.1 Mathematical model of speech production system [1]**

The model shown in Figure 1.1 is often called the LPC model. The speech signal comes form the LPC filter and the input to the model is either a train of pulses or a white noise sequence. In the case of physical model speech comes form vocal tract. Here vocal tract acts as a digital filter.

**Table 1.1 The relationship between the physical and the mathematical model of speech production system**

| Physical Model | Mathematical Model |
|---|---|
| Vocal Tract | H(z) (LPC filter) |
| Air | u(n) (innovations) |
| Vocal Cord Vibration | V (voiced) |
| Vocal Cord Vibration Period | T (pitch period) |
| Fricatives and Plosives | UV (unvoiced) |
| Air Volume | G (gain) |

## 1.4 Characteristics of Speech Signals

Deterministic signals could be described with the help of analytical formulas. But speech signal is not a deterministic signal. It is a random signal [2]. It can't be described by analytical formulas. In the speech we have two types of sounds one is voiced sounds and another is unvoiced sounds. These sounds are coming due to air compression in the lungs. When generating voiced sounds the vocal cards vibrate and generate a high energy quasi periodic speech waveform. Where as in the case of unvoiced sounds the vocal cards do not participate in the voice production and these unvoiced sounds behaves like noise.

In comparison to audio signals, speech signals can be characterized by a rater low analogue bandwidth. In standard communications applications a telephone bandwidth of 0.3 – 3.4 kHz allows a digital representation at a sampling frequency of 8 kHz [2]. In the case of audio signals like music has bandwidth of about 15 – 20 kHz and thus require a sampling frequency of 32 to 48 kHz. In between, wideband speech signals (bandwidth 7 kHz, sampling rate 16 kHz) have been attracting an increasing interest with reference to high quality ISDN applications, such as comfort telephony or videoconferencing services [3]. The typical waveforms of speech signal, voiced and unvoiced frames are shown in the following Figures 2.4, 2.5 and 2.6 respectively.

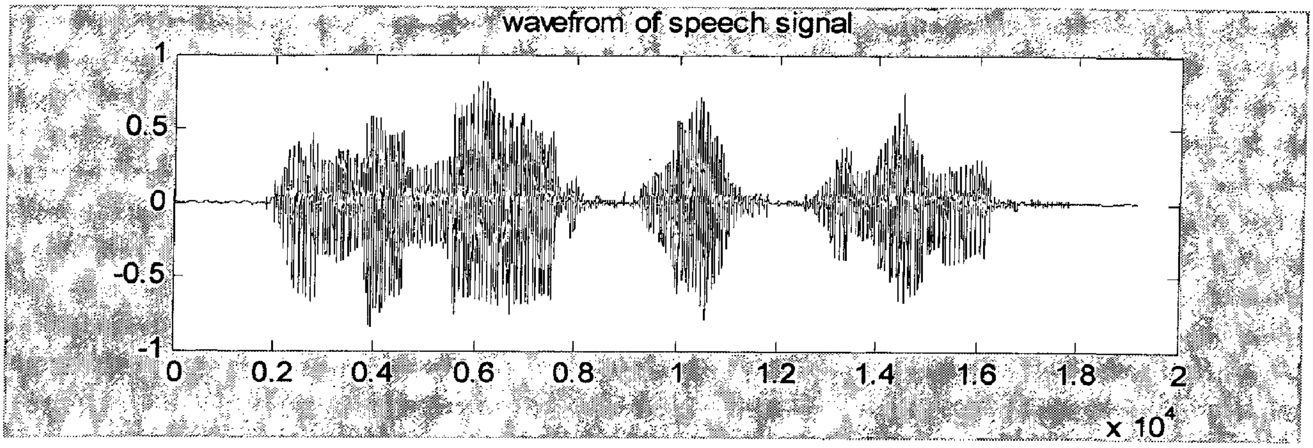


**Figure 1.2 Typical waveform of speech signal**

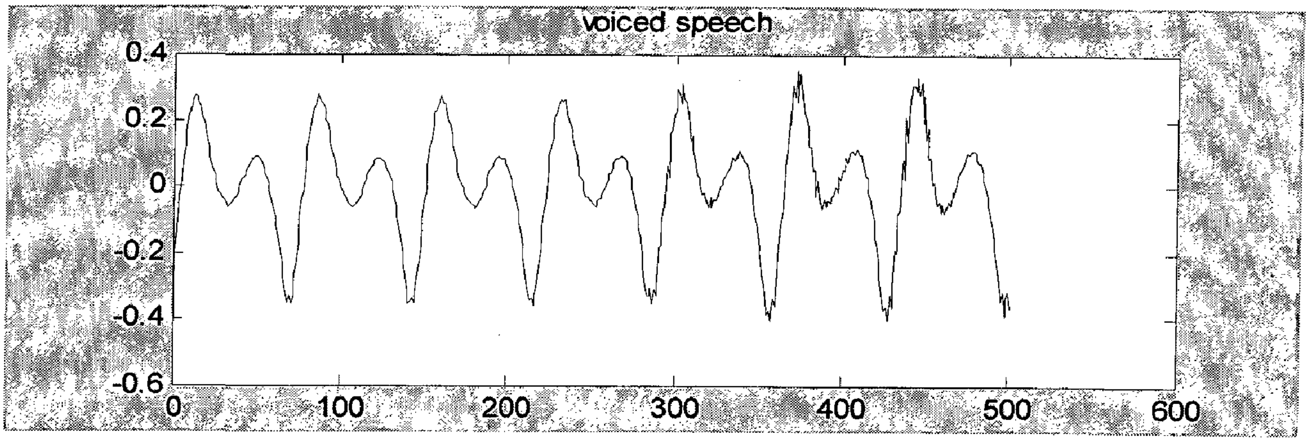


**Figure 1.3 Typical waveform of voiced speech**

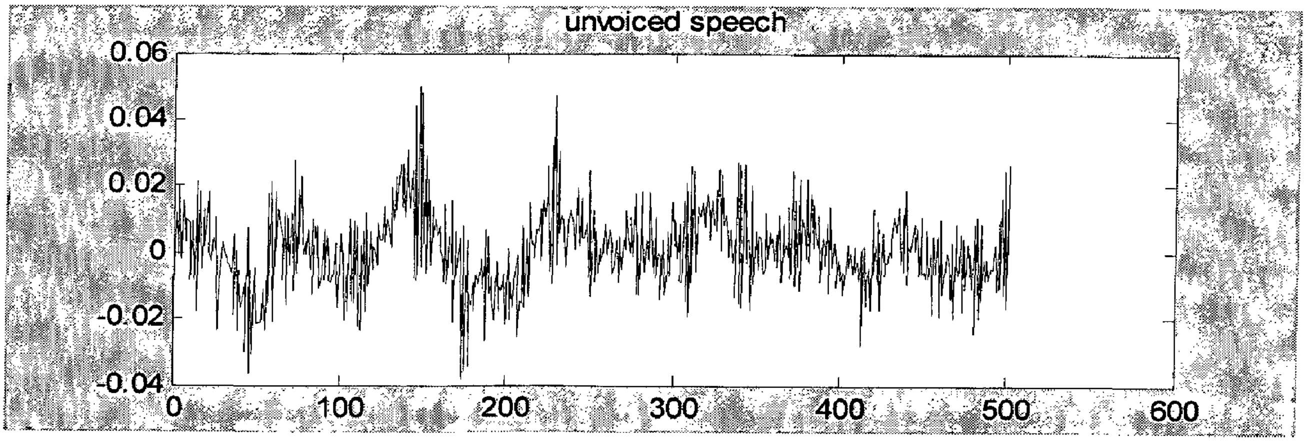


**Figure 1.4 Typical waveform of unvoiced speech**

## 1.5 Applications of Speech Compression

1. Speech compression is required in long distance communication. For example, in digital cellular technology many users need to share the same frequency bandwidth. So by utilizing speech compression makes it possible for more users to share the available system.

2. Another application where speech compression is need is in digital voice storage. For a fixed amount of available memory, compression makes it possible to store longer messages.
3. Fixed and mobile digital telephony.
4. Packet network transmission (internet).
5. Videoconferencing.
6. Radio and television.
7. Message encryption.

### 1.6 Organization of the Dissertation Report

The dissertation has been composed of seven chapters. The organization of this dissertation report is as follows.

In chapter 2, classification of speech compression techniques has been given. Properties of speech coding methods such as quality, bitrate (compression), delay, and complexity of speech coders has been explained. It has also been explained briefly about the different coding methods such as waveform coding, vocoding, and hybrid coding.

In chapter 3, waveform coding methods such as pulse code modulation, differential pulse code modulation and adaptive DPCM has been explained. Quantization and companding has also been explained in this chapter.

In chapter 4, LPC coding method has been explained. In this LPC coding method LPC analysis, Levinson-Durbin algorithm, correlation of signals, and pitch detection methods has been described.

In chapter 5, basics of wavelet transform has been explained. Different steps involved in the implementation of wavelet transform based speech compression are explained one by one.

In chapter 6, what ever the algorithms that are proposed in the present work, LPC coding, ADPCM and WT technique, simulation results are tabulated. For the analysis purpose different speech signals are taken form different persons. Comparative evaluation has been done on the basis of compression ratio, SNR, PSNR, and NRMSE.

Lastly, in Chapter 7, conclusion of present work and some suggestions for future work have been given.

## 2.1 Introduction

The objective of speech is communication whether face to face or cell phone to cell phone. To fit a transmission channel or storage space, speech signals are converted to formats using various techniques [2]. This is called speech coding or compression. Theoretically speaking, speech coding can be achieved based on two facts. One is redundancy in speech signals, and another one is perception properties of human ears. In this chapter we will discuss about classification of speech compression techniques.

Properties of speech coding techniques are discussed briefly one by one. Speech quality of compressed speech signal depends on compression ratio, complexity, delay, and bandwidth. These are nothing but the attributes of speech coders. So there is an interaction between all these attributes and that they can be traded off against each other. For example, in the case of low bit rate coders delay is more with respect to high bit rate coders and also complexity is more in the case of low bit rate coders and also quality is low in the case of low bit rate coders.

Different coding methods are discussed briefly. Waveform coding methods does not use the properties of the speech production model. These methods try to reproduce the output as close as to the original signal, so these methods are having the high quality. In the case of vocoding methods these use the properties of the speech production model. These give less quality output but compression is more. Hybrid coding methods use the advantages in both waveform coding and vocoding, so these are moderate in both quality as well as compression.

## 2.2 Classification of Speech Codecs

Speech coding methods are classified into three categories [2], [4].

1. waveform coding
2. vocoding
3. hybrid coding

The basic difference between speech codecs are shown in the Figure 2.1 [4]. In this speech quality versus bitrate performance of these codecs are shown. The bit rate is plotted on a logarithmic axis and the speech quality classes “poor to excellent” on another axis.

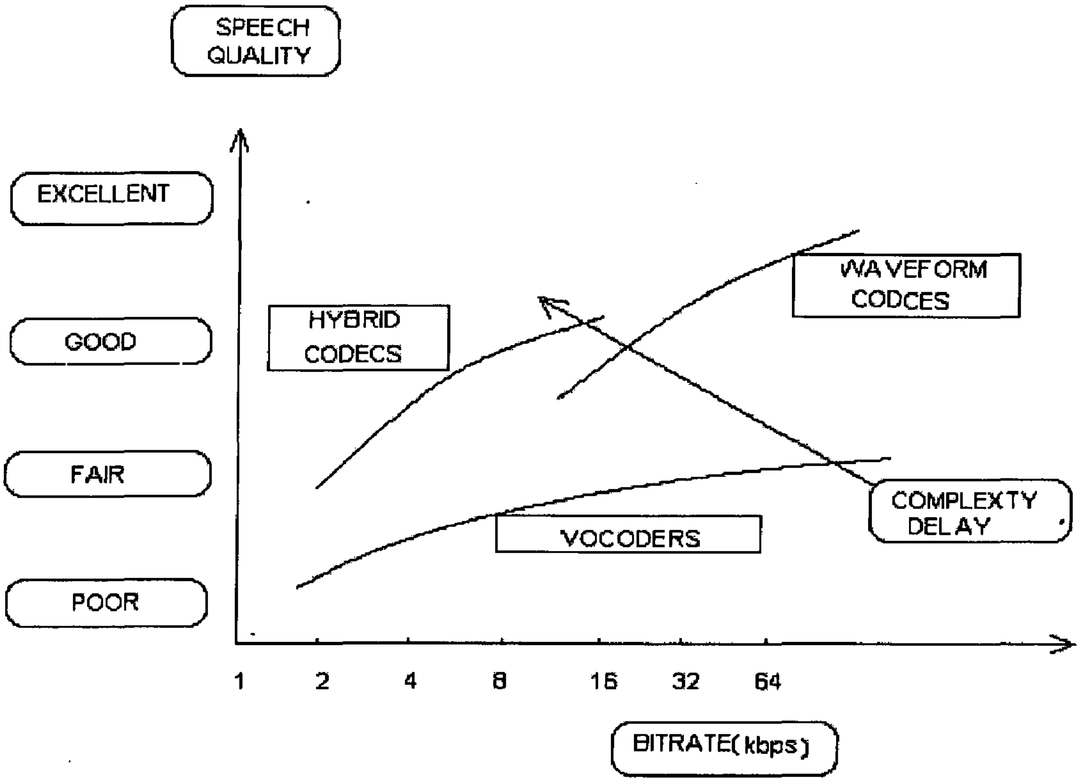


**Figure 2.1 Speech quality vs. bitrate classification of speech codecs.**

### 2.3 Attributes of Speech Coders [2], [3]

Speech quality of compressed speech signal depends on bit rate, complexity, delay, and bandwidth. These are nothing but the attributes of speech coders. So there is an interaction between all these attributes and that they can be traded off against each other. For example, in the case of low bit rate coders delay is more with respect to high bit rate coders and also complexity is more in the case of low bit rate coders and also quality is low in the case of low bit rate coders.

**Compression ratio:** It is important parameter in the case of speech compression because our main aim in speech compression is to reduce the size of the digital speech representation. Most speech coders operate at a fixed bit rate regardless of the input signal characteristics. Since multimedia speech coders share the channel with other forms of data so it is better to make the speech coder of variable bit rate. If the signal is declared speech, it is coded at the full fixed bit rate. If the signal is declared noise, it is coded at a lower bit rate. Sometimes no bits are transmitted at all. By using this type of algorithm we can reduces the bit rate

**Delay:** Delay of speech coding system is due to three reasons. One is due to processing of data frame of the speech coder. After process the speech parameters are updated and transmitted for every frame. Not only to analyze the data properly, it is necessary to analyze the data beyond the frame boundary. Hence, before the speech can be analyzed it is necessary to buffer a frame worth of data. Due to this, delay comes this delay is called an algorithmic delay. This type of delay we can't reduce in the implementation of the speech coder. So if we want to decrease the delay we have to go to other components of delay those depends on the implementation. It depends on the speed of the hardware used to implement the coder. The sum of the algorithmic and processing delays is called the one-way codec delay. The third component of delay is due to the communication delay, which is the time it takes for an entire frame of data to be transmitted from the encoder to the decoder. Total of these three delays are called as one way system delay.

**Complexity:** speech coders are often implemented on digital signal processor chips. The speed of digital signal processor is measured in terms of millions of instructions per second (MIPS). If the speech coder takes less MIPS then we can say that our speech coder is of less complex. Depending upon MIPS we can say the speech coder is complex or not and that's range is 15 MIPS to 30 MIPS. Lower limit is corresponding to lower complexity and higher limit is corresponding to higher complexity. Depending on the complexity, cost and power usage differs. If the complexity more means it takes more power, so due to that weight increases because it uses large size batteries. So complexity is important factor to be considered in the case of speech coders.

**Quality:** Even though our main aim is to compress the speech signal it is necessary to give the enough quality of speech. So compression and quality are equally important in the case of speech compression. In the case speech, ideal conditions may not get, that means in most of the cases speech contains noise. Noise may in the form of car noise, street noise, office noises like typing or phones ringing, air conditioning noise, music in the background, etc. Even in the case of noise conditions speech coder has to give enough quality of speech at the other end that means at the receiver end.

### 2.4 Waveform Coding

In general, waveform coding techniques are designed to be signal independent. That means waveform coding techniques treats speech signals as normal signal waveforms. Waveform coders then tries to obtain the most similar reconstructed s'(n) signal to the original one s(n) [2]. The objective of the waveform coding is to minimize the error e(n)=s'(n)-s(n). They are designed to map the waveform of the encoder into a facsimile-like replica of it at the output of the decoder. Because of this advantage, the waveform coding methods can also be used to encode secondary type of information such as signaling tones, voice band data, or even music. Because of the signal reproduction as accurately as possible the compression ratio is not good with respect to other coding methods. The coding efficiency can be improved by exploiting some statistical signal properties, if the codec parameters are optimized for the most likely categories of input signals, while still maintaining good quality for other types of signals as well. The waveform codecs can be further subdivided into time-domain waveform codecs and frequency-domain waveform codecs [4].

- **Time domain waveform coding.** The most well known representative of signal independent time domain waveform coding is the A-law companded pulse code modulation (PCM) scheme. A-law is a nonlinear type of companding. This results in near constant signal to noise ratio (SNR) over the total input dynamic range.
- **Frequency domain waveform coding.** In frequency domain waveform codecs, the input signal undergoes a more or less accurate short time spectral analysis. The signal is split into a number of sub bands, and the individual sub band signals are then encoded by using different numbers of bits in order to obey rate distortion theory on the basis of their prominence. Two well known representations of this class are sub band coding (SBC) and adaptive transform coding (ATC).

### 2.5 Vocoding

Vocoders is nothing but voice coders. Vocoders make no attempt to reproduce the original waveform as like waveform coding. Vocoders derive a set of parameters at the encoder which can be used to control a speech production model at the decoder. The parameter set for the speech production model is relatively small and can be efficiently quantized for transmission. The simplest model of speech production used by vocoders is illustrated in mathematical model of speech production system in Figure 1.1. The voiced signal is modeled by an unipolar pulses of unit amplitude impulses at the required fundamental frequency. Here the fundamental frequency is nothing but the pitch frequency. For unvoiced sounds the unvoiced excitation is modeled as the output form a pseudorandom noise generator. The voiced/unvoiced switch selects the appropriate excitation and the gain term controls the level of the excitation. So from the vocal tract filter synthesized speech comes out.

### 2.6 Hybrid Coding

Waveform coding methods are good with respect to speech quality and vocoding methods are good with respect to compression. By taking advantages in both the techniques hybrid coding methods are formed [5]. But hybrid coding methods have higher complexity. So every coding method formed by combining waveform coding and source coding methods falls under this category. Hybrid coding improves the speech quality and reduces the compression. Hybrid coding methods are often referred to as analysis by synthesis coding methods.

In the following Figure 2.2 classification of speech compression techniques has been given. But in this, classification of speech compression techniques has given differently [6]. In this classification frequency domain coding methods are categorized into hybrid coding methods.

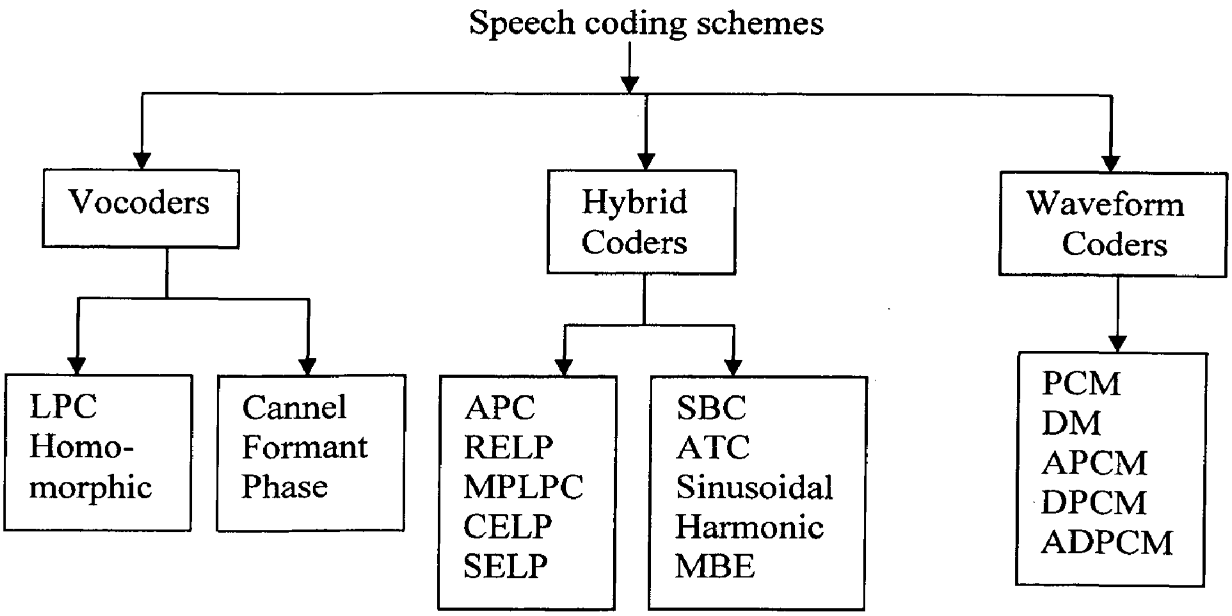


**Figure 2.2 Classification of speech coding schemes [6].**

# Chapter-3
# WAVEFORM CODING METHODS

## 3.1 Introduction

Waveform coding methods treats speech signals as normal signal waveforms. Then these methods try to obtain the most similar reconstructed signal to the original one. In general, waveform coding techniques are designed to be signal independent. Waveform coders then tries to obtain the most similar reconstructed s'(n) signal to the original one s(n). The objective of the waveform coding is to minimize the error e(n)=s'(n)-s(n). They are designed to map the waveform of the encoder into a facsimile-like replica of it at the output of the decoder. Because of this advantage, the waveform coding methods can also be used to encode secondary type of information such as signaling tones, voice band data, or even music. Because of the signal reproduction as accurately as possible the compression ratio is not good with respect to other coding methods. The coding efficiency can be improved by exploiting some statistical signal properties, if the codec parameters are optimized for the most likely categories of input signals, while still maintaining good quality for other types of signals as well.

In this chapter we will discuss about pulse code modulation, differential pulse code modulation and adaptive differential pulse code modulation briefly.

## 3.2 Pulse Code Modulation (PCM)

PCM is a waveform coding method defined in the ITU−T G.711 specification [7]. The following are the different steps that are involved in the pulse code modulation and in this section we will discuss these steps one by one.

### 3.2.1 Anti-aliasing Low Pass Filtering:

The first step to convert the signal from analog to digital is to filter out the higher frequency component of the signal. In case of speech signals, about 1% of the energy resides above 4 kHz and only a negligible proportion above 7 kHz. Even in the case of wide band speech systems the band limit is 7-8 kHz. Conventional telephone systems usually employ a bandwidth limitation of 0.3-3.4kHz, which results in a minor degradation, hardly perceptible by the untrained listener. So anti-aliasing low pass filtering is necessary in order to band limit the signal to a bandwidth of B before sampling.

### 3.2.2 Sampling:

The band limited speech is sampled according to the Nyquist Theorem. Which requires a minimum sampling frequency of $f_{Nyquist} = 2 \cdot B$. This process introduces time-discrete samples. Due to sampling, the original speech spectrum is replicated at multiples of the sampling frequency. This is why the previous band limitation was necessary in order to prevent aliasing or frequency domain overlapping of the spectral lobes. If this condition is met, the original analog speech signal can be restored from its samples by passing the samples through a lowpass filter (LPF) with a bandwidth of B. in conventional speech systems, typically a sampling frequency of 8 kHz corresponding to a sampling interval of 125 $\mu$ s is used.

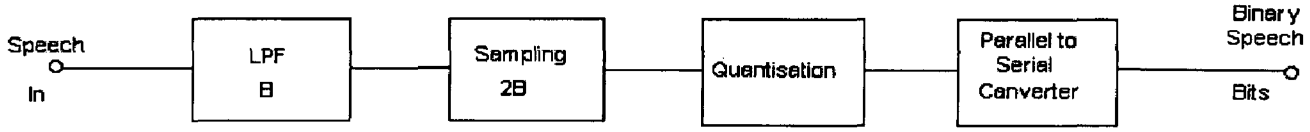


**Figure 3.1 Digitization of analogue speech signals [4].**

### 3.2.3 Quantization and Coding

Quantization is the process of converting each analog sample value into a discrete value that can be assigned a unique digital code word. All quantization intervals are equally spaced (uniform quantization) throughout the dynamic range of the input analog signal. Each quantization interval is assigned a discrete value in the form of a binary code word. The standard word size used is eight bits. If an input analog signal is sampled 8000 times per second and each sample is given a code word that is eight bits long, then the maximum transmission bit rate for Telephony systems using PCM is 64,000 bits per second. Each input sample is assigned a quantization interval that is closest to its amplitude height. If an input sample is not assigned a quantization interval that matches its actual height, then an error is introduced into the PCM process. This error is called quantization noise. Quantization noise is equivalent to the random noise that impacts the signal-to-noise ratio (SNR) of a voice signal. SNR is a measure of signal strength relative to background noise. The ratio is usually measured in decibels (dB). If the incoming signal strength in microvolts is Vs and the noise level also in microvolts is $V_n$ then the signal-to-noise ratio (SNR) in decibels is given by the formula S/N = 20 log10 ($V_s/V_n$). SNR is measured in decibels (dB). Higher the SNR better the voice quality. Quantization noise reduces the SNR of a signal. Therefore, an increase in quantization noise degrades the quality of a voice signal.

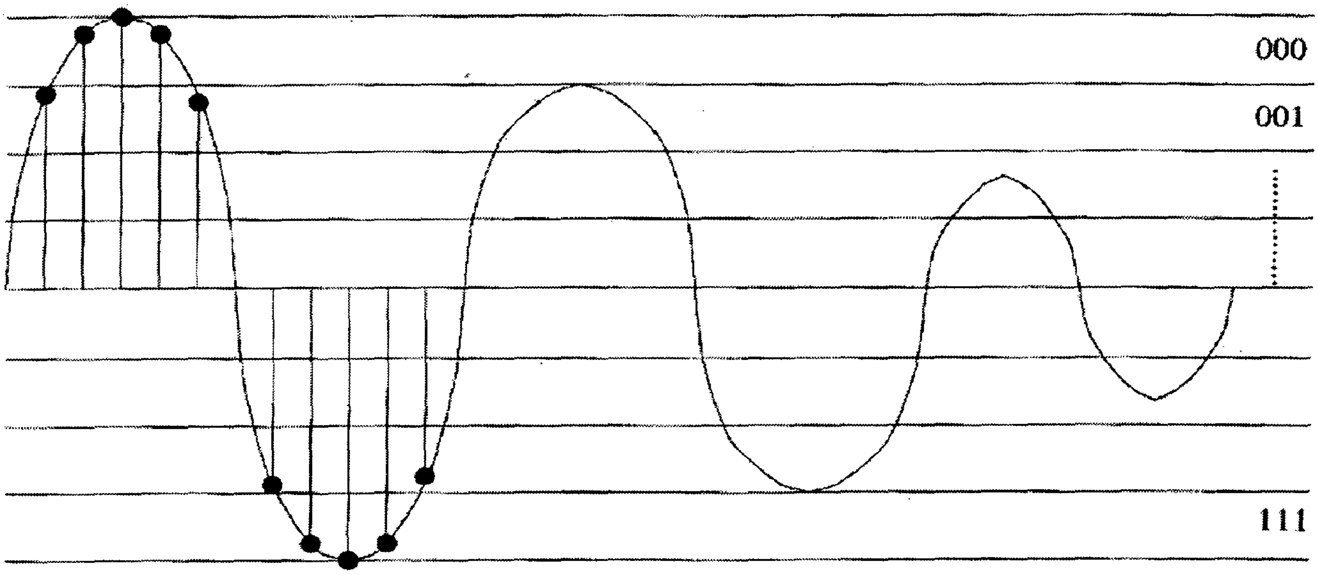


**Figure 3.2 Sampled and quantized analog signal [16].**

One way to reduce quantization noise is to increase the amount of quantization intervals. The difference between the input signal amplitude height and the quantization interval decreases as the quantization intervals are increased (increases in the intervals decrease the quantization noise). However, the amount of code words also need to be increased in proportion to the increase in quantization intervals. This process introduces additional problems that deal with the capacity of a PCM system to handle more code words.

The difference between the uniform and nonuniform quantizer characteristics is shown in Figure 3.3. SNR (including quantization noise) is the single most important factor that affects voice quality in uniform quantization. Uniform quantization uses equal quantization levels throughout the entire dynamic range of an input analog signal. Therefore, low signals have a small SNR (low-signal-level voice quality) and high signals have a large SNR (high-signal-level voice quality). Since most voice signals generated are of the low kind, having better voice quality at higher signal levels is a very inefficient way of digitizing voice signals. To improve voice quality at lower signal levels, uniform quantization (uniform PCM) is replaced by a nonuniform quantization process called companding.

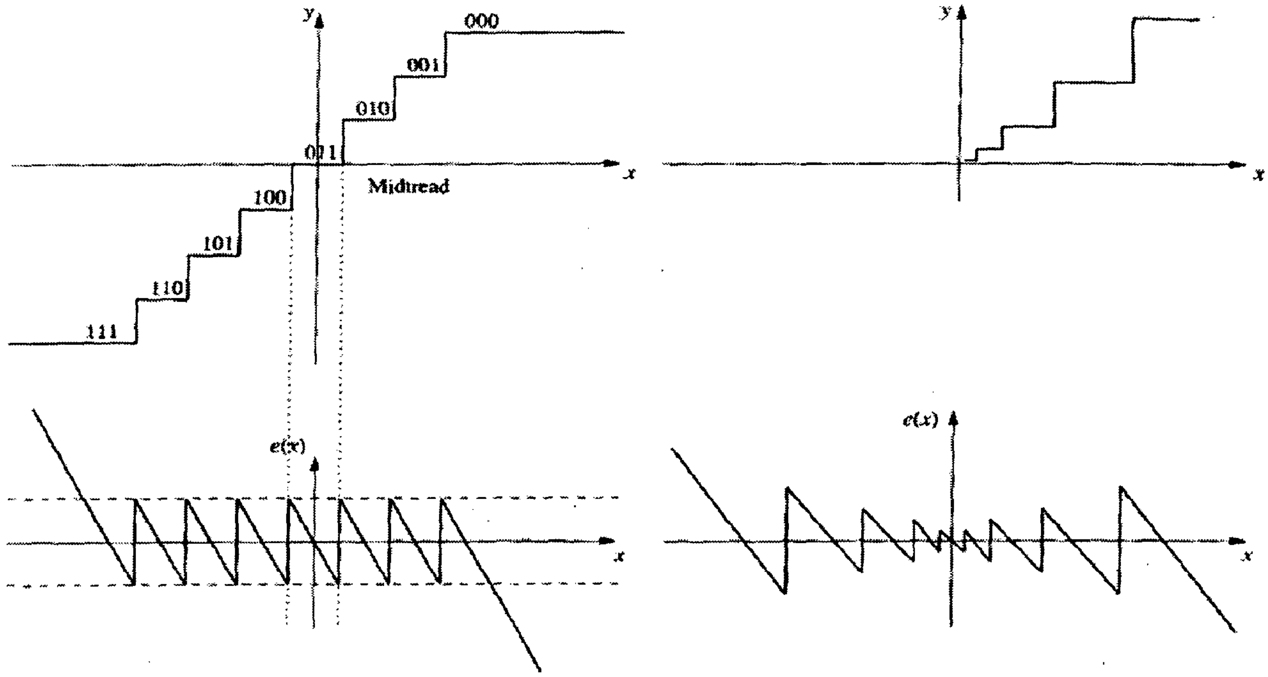


**Figure 3.3 Uniform and nonuniform quantizer characteristics**

### 3.2.4 Companding

Companding refers to the process of first compressing an analog signal at the source, and then expanding this signal back to its original size when it reaches its destination. The term companding is created by combining the two terms, compressing and expanding, into one word. At the time of the companding process, input analog signal samples are compressed into logarithmic segments. Each segment is then quantized and coded using uniform quantization. The compression process is logarithmic [4]. The compression increases as the sample signals increase. In other words, the larger sample signals are compressed more than the smaller sample signals. This causes the quantization noise to increase as the sample signal increases. A logarithmic increase in quantization noise throughout the dynamic range of an input sample signal keeps the SNR constant throughout this dynamic range

### 3.2.5 A−law and $\mu$−law Companding

Two practical logarithmic companders are one A-law compander and another one is $\mu$-law compander. A-law and $\mu$-law are audio compression schemes (codecs) defined by Consultative Committee for International Telephony and Telegraphy

(CCITT) G.711 which compress 16–bit linear PCM data down to eight bits of logarithmic data [8].

- **The $\mu$-law compander :** This companding characteristic is given by:

$$y = C(x) = y_{\max} \cdot \frac{\ln[1+\mu \cdot (|x|/x_{\max})]}{\ln(1+\mu)} \cdot \mathrm{sgn}(x) \text{ ............... } 3.1$$

upon inferring form the log(1+z) function that

$$\log(1+z) \approx z \qquad \text{if } z << 1, \qquad \text{.....................}3.2$$

in the case of small and large signals, respectively, we have from the equation 3.1 that:

$$y = C(x) = \begin{cases} y_{\max} \cdot \dfrac{\mu \cdot (|x|/x_{\max})}{\ln \mu} & if\, \mu \cdot \left(\dfrac{|x|}{x_{\max}}\right) \ll 1 \\ y_{\max} \cdot \dfrac{\ln[\mu \cdot (|x|/x_{\max})]}{\ln \mu} & if\, \mu \cdot \left(\dfrac{|x|}{x_{\max}}\right) \gg 1 \end{cases} \text{ ..................}3.3$$

which is a linear function of the normalized input signal $x/x_{\max}$ for small signals and a logarithmic function for large signals. The $\mu \cdot (|x|/x_{\max}) = 1$ value can be considered to be the breakpoint between the small- and large-signal operation, and the $|x| = x_{\max}/\mu$ is the corresponding abscissa value. In order to emphasize the logarithmic nature of the characteristic, $\mu$ must be large, which reduces the abscissa value of the beginning of the logarithmic section. The optimum value of $\mu$ may be dependent on the quantizer resolution R, and for R=8 the American standard pulse code modulation (PCM) speech transmission system recommends $\mu$=255.

- **The A-law compander:** Another practical logarithmic compander characteristic is the A-law compander.

$$y = C(x) = \begin{cases} y_{\max} \cdot \dfrac{A(|x|/x_{\max})}{1+\ln A} \cdot \mathrm{sgn}(x); & 0 < \dfrac{|x|}{x_{\max}} < \dfrac{1}{A} \\ y_{\max} \cdot \dfrac{1+\ln[A(|x|/x_{\max})]}{1+\ln A} \cdot \mathrm{sgn}(x); & \dfrac{1}{A} < \dfrac{|x|}{x_{\max}} < 1 \end{cases} . \text{ ..................}3.4$$

where A=87.56. Similarly to the $\mu$-law characteristic, it has a linear region near the origin and a logarithmic section above the breakpoint $|x| = x_{\max}/A$. Note, however, that in case of R=8 bits $A<\mu$, hence, the A-law characteristic's linear-logarithmic breakpoint is at a higher input value than that of the $\mu$ -law characteristic.

### 3.3 Differential Pulse Code Modulation

At the time of the PCM process, the differences between input sample signals are minimal. Differential PCM (DPCM) is designed to calculate this difference and then transmit this small difference signal instead of the entire input sample signal [9]. Since the difference between input samples is less than an entire input sample, the number of bits required for transmission is reduced. This allows for a reduction in the throughput required to transmit voice signals. Using DPCM can reduce the bit rate of voice transmission down to 48 kbps.

How does DPCM calculate the difference between the current sample signal and a previous sample? The first part of DPCM works exactly like PCM (that is why it is called differential PCM). The input signal is sampled at a constant sampling frequency (twice the input frequency). Then these samples are modulated using the PAM process. At this point, the DPCM process takes over. The sampled input signal is stored in what is called a predictor. The predictor takes the stored sample signal and sends it through a differentiator. The differentiator compares the previous sample signal with the current sample signal and sends this difference to the quantizing and coding phase of PCM (this phase can be uniform quantizing or companding with A-law or μ-law). After quantizing and coding, the difference signal is transmitted to its final destination. At the receiving end of the network, everything is reversed. First the difference signal is dequantized. Then this difference signal is added to a sample signal stored in a predictor and sent to a low-pass filter that reconstructs the original input signal.

DPCM is a good way to reduce the bit rate for voice transmission. However, it causes some other problems that deal with voice quality. DPCM quantizes and encodes the difference between a previous sample input signal and a current sample input signal. DPCM quantizes the difference signal using uniform quantization. Uniform quantization generates an SNR that is small for small input sample signals and large for large input sample signals. Therefore, the voice quality is better at higher signals. This scenario is very inefficient, since most of the signals generated by the human voice are small. Voice quality needs to focus on small signals. To solve this problem, adaptive DPCM is developed.

### 3.4 Adaptive Differential Pulse Code Modulation

Adaptive differential pulse code modulation (ADPCM) codecs are waveform codecs which instead of quantizing the speech signal directly, like PCM codecs, quantize the difference between the speech signal and a prediction that has been made of the speech signal. If the prediction is accurate then the difference between the real and predicted speech samples will have a lower variance then the real speech samples, and will be accurately quantized with fewer bits than would be needed to quantize the original speech samples [9]. At the decoder the quantized difference signal is added to the predicted signal to give the reconstructed speech signal.

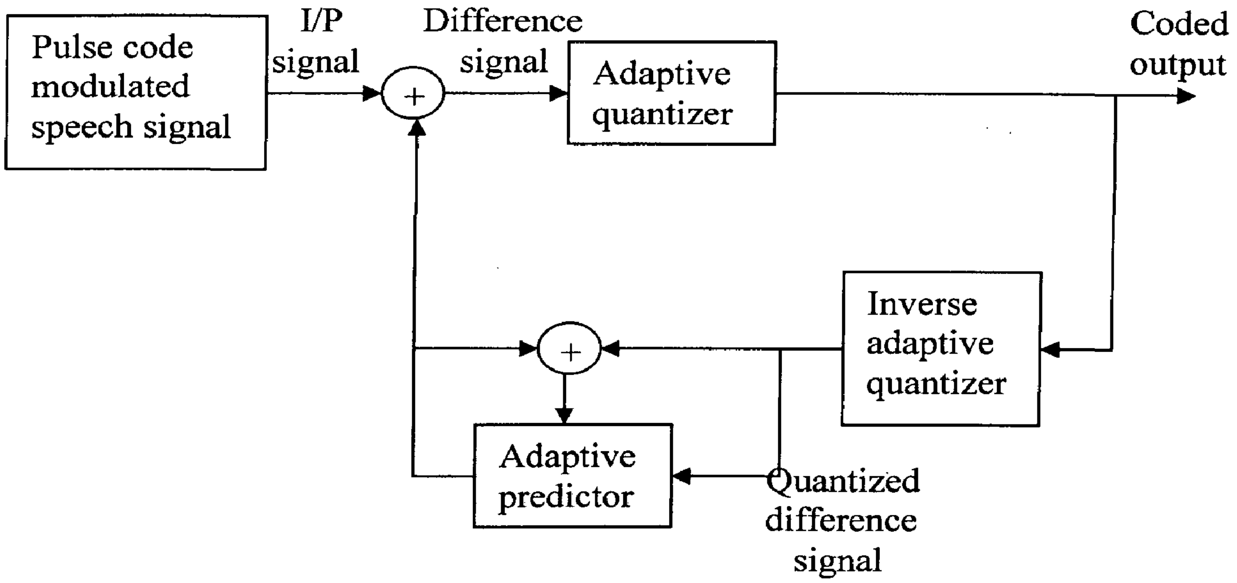

Pulse code modulated speech signal
I/P signal
+
Difference signal
Adaptive quantizer
Coded output
Inverse adaptive quantizer
+
Adaptive predictor
Quantized difference signal

**Figure 3.4 Detailed ADPCM encoder schematic.**

In the ADPCM technique ADPCM adapts the quantization levels of the difference signal that generated at the time of the DPCM process. How does ADPCM adapt these quantization levels?. If the difference signal is low, ADPCM increases the size of the quantization levels. If the difference signal is high, ADPCM decreases the size of the quantization levels. So, ADPCM adapts the quantization level to the size of the input difference signal. This generates an SNR that is uniform through out the dynamic range of the difference signal. So by using ADPCM we can reduce the size of the input speech signal.

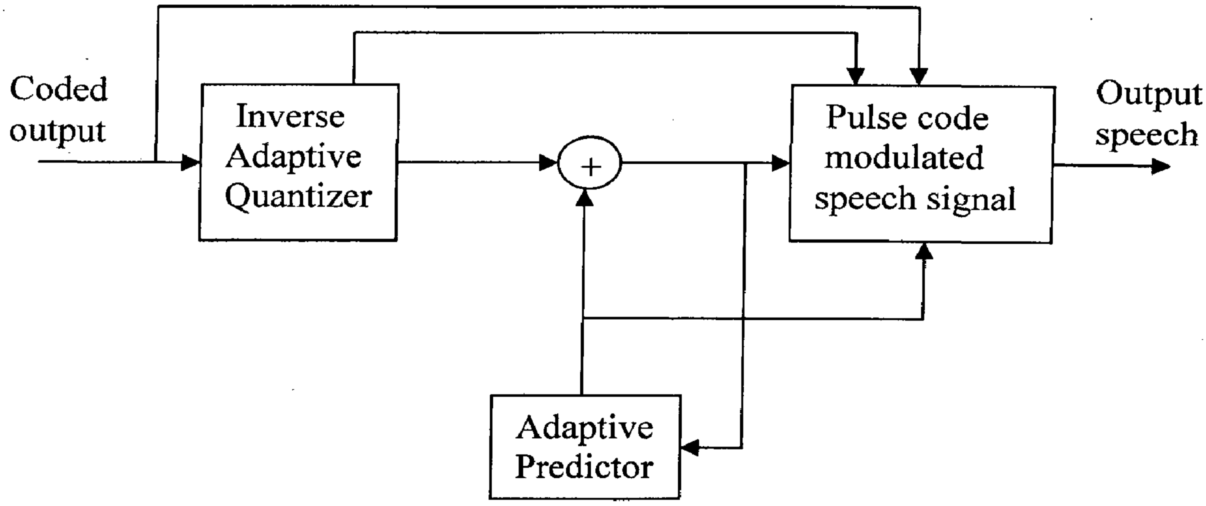

Coded output
Inverse Adaptive Quantizer
+
Pulse code modulated speech signal
Output speech
Adaptive Predictor

**Figure 3.5 Detailed ADPCM decoder schematic.**

ADPCM is the advancement of DPCM coding intern advanced to the PCM. So pulse code modulated speech signal is given as an input to the differentiator. So the input signal's estimate produced by the adaptive predictor is subtracted form the input in order to produce a difference signal having a lower variance. This lower variance difference signal can then be adaptively quantized with lower noise variance than the original signal with the help of lower bits than that of direct PCM method. This encoded speech signal is then transmitted to the decoder part of ADPCM. Furthermore, it is locally decoded, using the inverse adaptive quantizer to deliver the locally reconstructed quantized difference signal. This signal is added to the previous signal estimate in order to yield the locally reconstructed signal. Based on the quantized difference signal and the locally reconstructed signal, the adaptive predictor derives the subsequent signal estimate, and so on.

# Chapter – 4
# VOCODERS

## 4.1 Introduction

A second form of source coding for speech compression is provided by vocoders. These devices extract the characteristic parameters of human speech, by analyzing the mechanisms of speech formation, to derive an algorithm for providing additional compression of the data to be transmitted to the receiver. Recent results showed that the data rate of speech could be compressed to less than 2.4 kbps by using linear predictive coding (LPC) vocoders [10]. We know that air compressed by the lungs excites the vocal cords in two typical modes. The excitation signal denoted by E(z) in z-domain is then filtered through the vocal apparatus, which behaves like a spectral shaping filter with a transfer function of H(z)=1/A(z) that is constituted by the spectral shaping action of the glottis, vocal tract, lip radiation characteristics, and so on.

Accordingly, instead of attempting to produce a close replica of the input signal at the output of the decoder, the appropriate set of source parameters is found in order to characterize the input signal sufficiently closely for a given duration of time. First, a decision must be made as to whether the current speech segment to be encoded is voiced or unvoiced. Then the corresponding source parameters must be specified [11]. In the case of voiced sounds, the source parameter is the time between periodic vocal tract excitation pulses, which is often referred to as the pitch p. In the case of unvoiced sounds, the variance or power of the noise-like excitation must be determined. The parameters are quantized and transmitted to the decoder in order to synthesize a replica of the original signal. Vocoder schematic is shown in the Figure 4.1. The encoder is a simple speech analyzer, determining the current source parameters. After initial speech segmentation, it computes the linear predictive filter coefficients $a_i$ i=1....p, which characterize the spectral shaping transfer function H(z). A voiced/unvoiced decision is carried out, and the corresponding pitch frequency and noise energy parameters are determined [2]. These are then quantized, multiplexed, and transmitted to the speech decoder, which is a speech synthesizer.

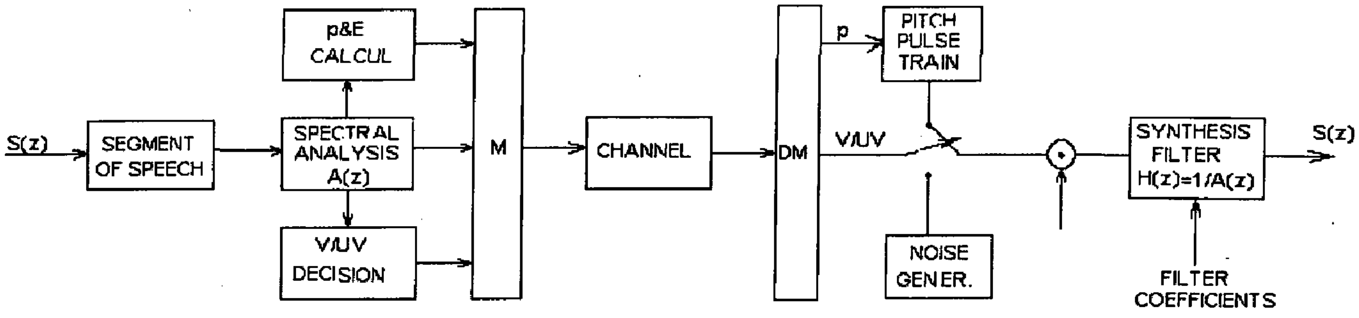


**Figure 4.1 Vocoder schematic.**

The associated speech quality of this type of systems may be predetermined by the adequacy of the source model, rather than by the accuracy of the quantization of these parameters. In linear predictive coding (LPC), often more complex excitation models are used to describe the voice-generating source. Once the vocal apparatus has been described by the help of its spectral domain transfer function H(z), the central problem of coding is to decide how to find the simplest adequate excitation for high quality parametric speech representation.

### 4.2 LPC Analysis

In the linear prediction signal is modeled as a linear combination of its past values and present and past values of a hypothetical input to a system whose output is the given signal [12]. Linear predictive coding is a standard model for speech coders. In this model all-pole model is used to describe the transfer function of the vocal tract. Here it is showing the procedure to get the synthetic speech [12]. Assume that the present sample of the speech is predicted by the past $P$ samples of the speech such that

$$S'(n) = \sum_{i=1}^{p} a_i S(n-i) \qquad \text{.....................4.1}$$

where $S'(n)$ is the prediction of $S$ (n), $S$ *(n-k)* is the $k^{th}$ step previous sample, and $\{a_i\}$ are the linear prediction coefficients. The residual error between the actual sample and the predicted one can be expressed as

$$\text{e(n)}= S\text{ (n)-} S'(n) = S\text{ (n)-} \sum_{i=1}^{p} a_i S(n-i) \qquad \text{............... 4.2}$$

The sum of the squared error to be minimized is expressed as

$$E=\sum_{n}e^{2}(n)=\sum_{n}\left(S(n)-\sum_{i=1}^{p}a_{i}S(n-i)\right)^{2} \qquad \text{..............} \ 4.3$$

We would like to minimize the sum of the squared error. By setting to zero the derivative of $E$ with respect to $a_i$, one obtains

$$2\sum_{n}S(n-k)\left(S(n)-\sum_{i=1}^{p}a_{i}S(n-i)\right)=0$$

for k=1,2,3 ····· p. .............. 4.4

$$E\{S(n)S(n-k)\}=E\left\{\sum_{i=1}^{p}a_{i}S(n-i)S(n-k)\right\} \qquad \text{..............} 4.5$$

Upon exchange the order of the summation and expected value computation at the right hand side of equation 4.5 we get following equation.

$$E\{S(n)S(n-k)\}=\sum_{i=1}^{p}a_{i}E\{S(n-i)S(n-k)\} \quad \text{k=1,. . . p} \qquad \text{. ..........}4.6$$

By observing the above equation

$$C(k,i)=E\{S(n-k)S(n-i)\} \qquad \text{..........}4.7$$

Equation 4.7 represents the input signal's covariance coefficients. The covariance coefficients C(k, i) are now computed form the following short-term expected value expression:

$$C(k,i)=\sum_{n=0}^{L_a+p-1}S(n-k)S(n-i), \quad {}^{k=1,....p,}_{i=1,.....p.} \qquad \text{..............}4.8$$

Upon setting m=n-k, equation 3.8 can be expressed as

$$C(k,i)=\sum_{m=0}^{L_a-1-(k-i)}S(m)S(m+k-i) \qquad \text{.............}4.9$$

$$\sum_{i=1}^{p}a_{i}C(k,i)=C(i,0) \quad \text{i=1,.......p} \qquad \text{. .........}4.10$$

Which suggests that C(k,i) is the short-time autocorrelation of the input signal s(m) evaluated at a displacement of (k-i), giving:

C(k,i)=r(k-i), ..........4.11

Where

$$r(j)=\sum_{n=0}^{L_a-1-j}S(n)S(n+j)=\sum_{n=j}^{L_a-1}S(n)S(n-j) \qquad \text{..........}4.12$$

Where r(j) represents the speech autocorrelation coefficients. Then the set of p equations () can now be reformulated as

$$\sum_{i=1}^{p} a_i r(|k-i|) = r(i) \quad \text{k=1,....,p} \qquad \text{.................4.13}$$

Equation (4) results in *P* unknowns in *P* equations.

The speech signal is divided into segments each with *N* samples. If the length of each segment is short enough, the speech signal in the segment may be stationary. In other words, the vocal tract model is fixed over the time period of one segment. If there are *N* samples in the sequence indexed from 0 to N-1 such that $\{S(n)\} = \{S(0), S(1), S(2), \cdots\cdots S(N-2), S(N-1)\}$, Equation () can be expressed in terms of matrix equation.

$$\begin{bmatrix} r(0) & r(1) & \cdots\cdot & r(p-1) \\ r(1) & r(0) & \cdots\cdot & r(p-2) \\ \cdot & \cdot & & \cdot \\ \cdot & \cdot & & \cdot \\ r(p-1) & r(p-2) & \cdots\cdot & r(0) \end{bmatrix} \begin{bmatrix} a_1 \\ a_2 \\ \cdot \\ \cdot \\ a_p \end{bmatrix} = \begin{bmatrix} r(1) \\ r(2) \\ \cdot \\ \cdot \\ r(p) \end{bmatrix}$$

$$\text{R} \qquad \text{a} = \text{r} \qquad \text{.......... 4.14}$$

where
$$r(k) = \sum_{n=0}^{N-1-k} S(n)S(n-k) \qquad \text{............. 4.15}$$

to solve the matrix equation 3.13 any of the following methods are used [1].

- The Gaussian elimination method.
- Any matrix inversion method (MATLAB).
- The Levinson-Durbin recursion.

out of those three algorithms Levinson-Durbin algorithm gives better results. Here the Levinson-Durbin algorithm has been given.

$$E(0) = r(0)$$

For k=1 to *p* do

$$i_k = \left[ r(k) - \sum_{j=1}^{k-1} a_j^{(k-1)} r(k-j) \right] / E(k-1) \qquad \text{.............4.16}$$

$$a_k^{(k)} = i_k$$

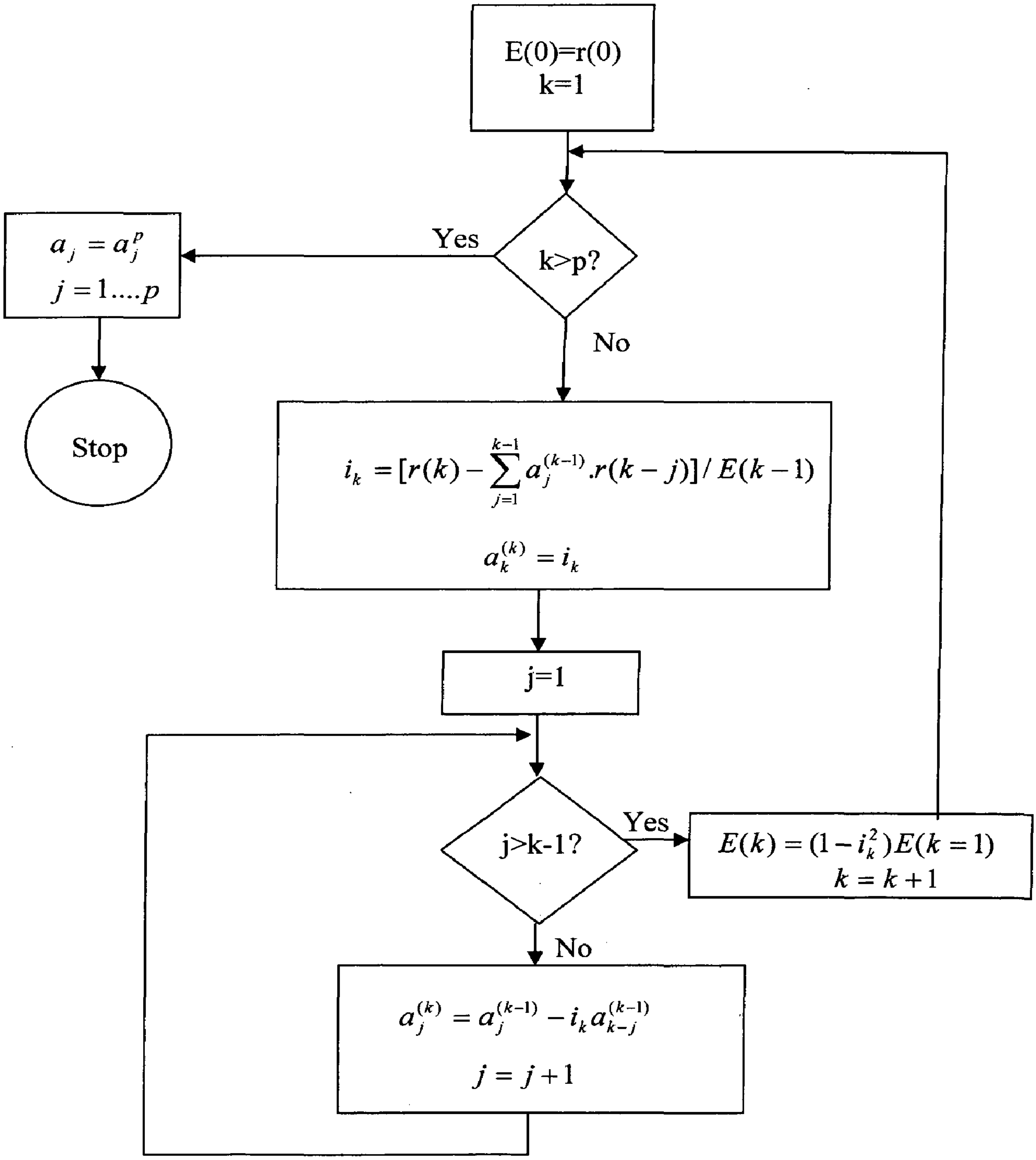
For j=1 to k-1 do

$$a_j^{(k)} = a_j^{(k-1)} - i_k a_{k-j}^{(k-1)} \qquad \text{............4.17}$$

$$E(i) = (1 - i_k^2)E(k-1). \qquad \text{............4.18}$$

The final solution after p iterations is given by:

$$a_j = a_j^{(p)} \qquad j=1,\dots\dots,p. \qquad \text{............4.19}$$



E(0)=r(0)
k=1
k>p?
Yes
No
$a_j = a_j^p$
$j = 1....p$
Stop
$i_k = [r(k) - \sum_{j=1}^{k-1} a_j^{(k-1)}.r(k-j)] / E(k-1)$
$a_k^{(k)} = i_k$
j=1
j>k-1?
Yes
No
$E(k) = (1 - i_k^2)E(k=1)$
$k = k+1$
$a_j^{(k)} = a_j^{(k-1)} - i_k a_{k-j}^{(k-1)}$
$j = j+1$

**Figure 4.2 Flow chart of levenson algorithm [2].**

Once the linear prediction coefficients $\{a_i\}$ are computed, equation 4.2 can be used to compute the residual error sequence e(n). The implementation of equation 4.2, where s(n) is the input and e(n) is the output, is called the analysis filter and is shown in Figure 4.3

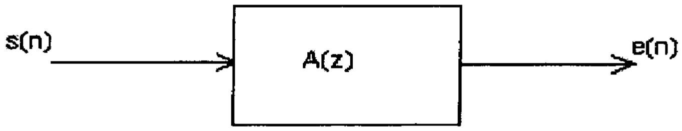


**Fig. 4.3 Speech analysis filter**

The transfer function is given by

$$A(z) = 1 - \sum_{i=1}^{p} a_i z^{-i} \qquad \text{.................. } 4.20$$

Because e(n) has less standard deviation than speech itself, smaller number of bits is needed to quantize the error sequence.

Equation 4.2 can be rewritten as the difference equation of a digital filter whose input is e(n ) and output is S(n) such that

$$S(n) = \sum_{i=1}^{p} a_i S(n-i) + e(n) \qquad \text{............}4.21$$

the implementation of Equation 4.21 is called the synthesis filter and is shown in Figure 4.4.

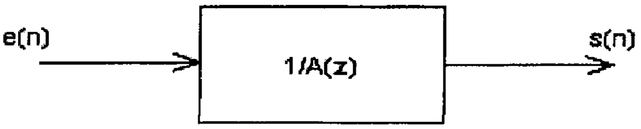


**Fig. 4.4 Speech synthesis filter**

If both the linear prediction coefficients and the error sequence are available, the speech can be reconstructed using the synthesis filter [15].

### 4.3 Correlation

The correlation is one of the most common and most useful statistics. A correlation is a single number that describes the degree of relationship between two variables. Correlation between groups of data implies that they move or change with respect to each other in a structured way. In the case of signals, signals have to be digitized and that therefore form groups of data. For N pairs of data {x(n), y(n)}, the coefficient is defined as

$$r_{xy} = \frac{\sum_{n=1}^{N}\{x(n)-\bar{x}\}\{y(n)-\bar{y}\}}{\sqrt{\sum_{n=1}^{N}\{x(n)-\bar{x}\}^2\sum_{n=1}^{N}\{y(n)-\bar{y}\}^2}} \qquad \text{.........} \quad 4.22$$

If the finite length signals are to be analyzed, then the definition of the crosscorrelation function of the two signals is as follows.

$$r_{xy}(k) = \frac{\sum_{n=1}^{N}\{x(n)-\bar{x}\}\{y(n+k)-\bar{y}\}}{\sqrt{\sum_{n=1}^{N}\{x(n)-\bar{x}\}^2\sum_{n=1}^{N}\{y(n)-\bar{y}\}^2}} \qquad \text{..........} \; 4.23$$

In the case when the two input signals are the same, the crosscorrelation function becomes the autocorrelation function of that signal. Thus, the autocorrelation function is defined as

$$r_{xx}(k) = \frac{\sum_{n=1}^{N}\{x(n)-\bar{x}\}\{x(n+k)-\bar{x}\}}{\sum_{n=1}^{N}\{x(n)-\bar{x}\}^2} \qquad \text{..........} 4.24$$

Autocorrelation is useful to determine the pitch period of the speech signals and also this is useful in the determination of voiced and unvoiced frames in the speech signals.

### 4.4 Pitch Detection

Pitch period is important factor in the case of voice coding methods. Accurate estimation of the pitch period or the lag in the pitch filter is very important. It is difficult to measure exact pitch period due to the following reasons [6]:

- The glottal excitation waveform is not a perfect train of periodic pulses. Although finding the period of a perfectly periodic waveform is straightforward, measuring the period of speech waveform, which varies both in period and in the detailed structure of the waveform within a period, can be quite difficult.
- In some instances, the formants of the vocal track can alter significantly the structure of the glottal waveform so that the actual pitch period is difficult to detect. Such interactions generally are most deleterious to pitch detection

during rapid movements of the articulators when the formants are also changing rapidly.

- The reliable measurement of pitch is limited by the inherent difficulty in defining the exact beginning and end of each pitch period during voiced speech segments.
- Another difficulty in pitch detection is distinguishing between unvoiced speech and low level voiced speech. In many cases, transitions between unvoiced speech segments and low level voiced speech segments are very subtle, and thus are extremely hard to pinpoint.
- In practical application, the background ambient noise can also seriously affect the performance of the pitch detector. This is especially serious in mobile communication environments where a high level of noise is present.

Pitch diction methods can be classified in the following categories [6]:

I. Pitch detectors which utilize the frequency domain properties of speech signals.

II. Pitch detectors which utilize the time domain properties of speech signals.

III. Pitch detectors which utilize both the frequency and time domain properties of speech signals.

In this dissertation work we are mainly interested in pitch detectors which utilize the time domain properties of speech signals. One major property of periodic signals is that the distant similarity of the waveform in time domain. The main principle of pitch detection algorithms (PDAs) which rely on waveform similarities is to find the pitch by comparing the similarity between the original signal and its shifted version. If the shifted distance is equal to the pitch, the two signal waveforms should have the greatest similarity. The majority of existing PDAs are based on this concept. Among them, the auto-correlation (AC) method and the average magnitude difference function (AMDF) are the two most widely used. Out of these two we are concentrating on only auto-correlation method.

The key problem of PDAs which are based on the waveform similarity methods is the quantitative definition of similarity [6]. There are a number of different similarity measures which result in different PDAs and performance. They are mainly based on the minimization of a quadratic cost function. The direct distance measurement is the most popular criterion, examining the similarity between two waveforms which can be expressed as

$$E(\tau) = \frac{1}{N}\sum_{n=0}^{N-1}[s(n) - s(n+\tau)]^2 \qquad \text{.........4.25}$$

where N is the analysis frame length and τ is the shifted distance. The above equation assumes that the average signal level is fixed. The assumption in the case of auto-correlation method is that the signal is stationary. The error criterion of equation (4.25) can be rewritten as

$$E(\tau) = [R(0) - R(\tau)] \qquad \text{..........4.26}$$

where

$$R(\tau) = \sum_{n=0}^{N-1} s(n)s(n+\tau) \qquad \text{.........4.27}$$

The minimization of the estimation error, E(τ), in equation 4.26 is equivalent to maximizing the auto-correlation R(τ). The variable τ is called lag or delay, and the pitch is equal to the value of τ, which results in the maximum R(τ).

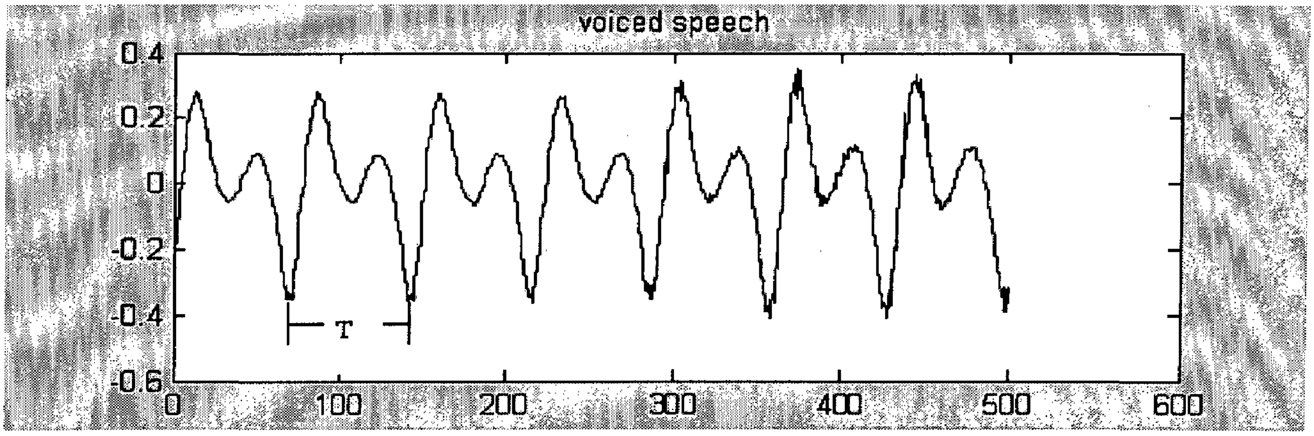


**Figure 4.5 Illustration of pitch period**

In the above Figure 4.5 has been shown that the pitch period (T) of the speech signal. Pitch period is a variable parameter. Its value changes from person to person's speech signal. In the case of women and young children the value of pitch period is more than that of men. Vocal card vibration determines the pitch period of the speech signal. So in the case of women and young children vibration of vocal card is faster than that of males.

### 4.5 Voiced/ Unvoiced Decision

By calculating the auto correlation coefficient we can say that the particular frame is voiced or unvoiced. Auto correlation coefficient can be calculated by using the equation (4.27). If the auto correlation coefficient is maximum only once in a frame then we can say that, that frame is unvoiced. If the auto correlation coefficient is giving maximum value repeatedly with particular interval then we can say that the frame is voiced with pitch period of T. Detection of voiced, unvoiced and pitch period is showing in the Figure (4.6).

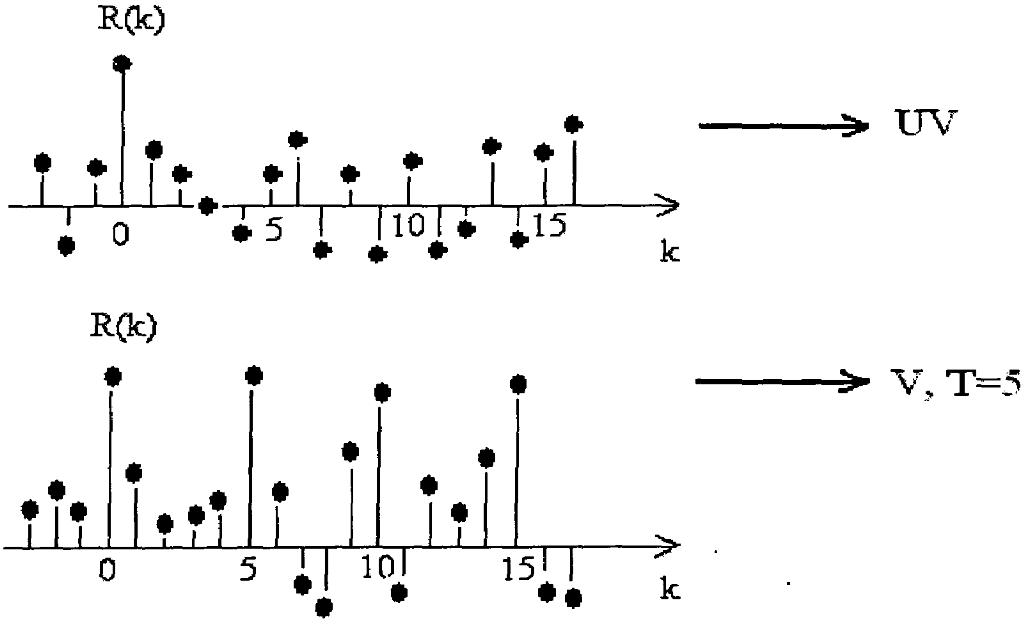


**Figure 4.6 Voiced/Unvoiced Decisions [1]**

### 5.1 Introduction

The fundamental idea behind wavelets is to analyze according to scale. The wavelet analysis procedure is to adopt a wavelet prototype function called an analyzing wavelet or mother wavelet. Any signal can then be represented by translated and scaled versions of the mother wavelet. Wavelet analysis is capable of revealing aspects of data that other signal analysis techniques such as Fourier analysis miss aspects like trends, breakdown points, discontinuities in higher derivatives, and self-similarity. Furthermore, because it affords a different view of data then those presented by traditional techniques, it can compress or de-noise a signal without appreciable degradation. Wavelets are functions that satisfy certain mathematical requirements and are used in representing data and other functions. However, in wavelet analysis, the scale that we use to look at data plays a special role. Wavelet algorithms process data at different scales or resolutions. If we look at a signal (or a function) through a large "window", we would notice gross features. Similarly, if we look at a signal through a small "window", we would notice small features.

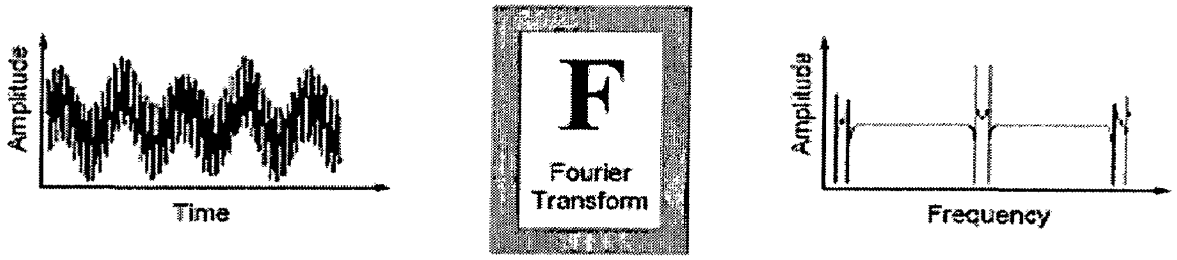
In this chapter we will discuss some background information on wavelets and wavelet transforms. About continuous wavelet transform and discrete wavelet transform will discuss. Different steps that are involved for the implementation of speech compression using wavelets will discuss.

### 5.2 Basics of Wavelet Transform

Transform techniques such as the discrete Fourier transform (DFT) or discrete cosine transform (DCT), and sub band techniques such as the conjugate quadrature-mirror filter bank (QMF) are suitable for stationary signal analysis. However, they are not suitable for analysis of non stationary signals such as speech and audio (time frequency) or images (space-frequency), or video (time-space frequency), and for nonlinear perceptual distortion criteria. Consequently, these techniques have recently been extended to QMF trees with unequal-bandwidth branches and sub band-DFT hybrids. A much more promising approach to time-frequency analysis is offered by wavelets [14].

**Fourier analysis:**

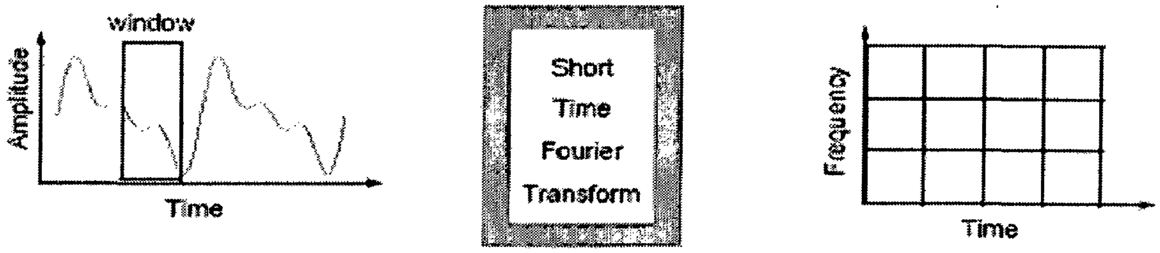
Fourier analysis break downs the signal into constituent sinusoids of different frequencies. So these sines and cosines are the basis functions and the elements of Fourier synthesis. It is nothing but, it transforms the time based signal into frequency based signal [15].



Fourier analysis has a serious drawback. In transforming to the frequency domain, time information is lost. When looking at a Fourier transform of a signal, it is impossible to tell when a particular event took place.

**Short-time Fourier analysis:**

In an effort to correct this deficiency Fourier analysis, Dennis Gabor (1946) adapted the Fourier transform to analyze only a small section of the signal at a time — a technique called windowing the signal. Gabor's adaptation, called the Short-Time Fourier Transform (STFT), maps a signal into a two-dimensional function of time and frequency.

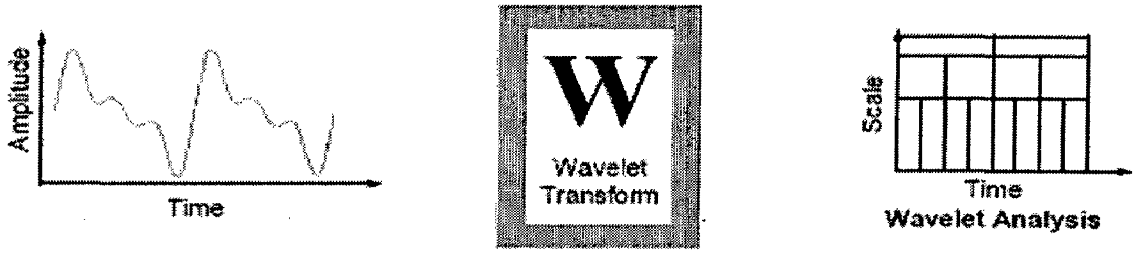


The STFT represents a sort of compromise between the time and frequency based views of a signal. It provides some information about both when and at what frequencies a signal event occurs. However, you can only obtain this information with limited precision, and that precision is determined by the size of the window. While the STFT compromise between time and frequency information can be useful, the drawback is that once you choose a particular size for the time window, that window is the same for all frequencies. Many signals require a more flexible approach — one where we can vary the window size to determine more accurately either time or frequency.

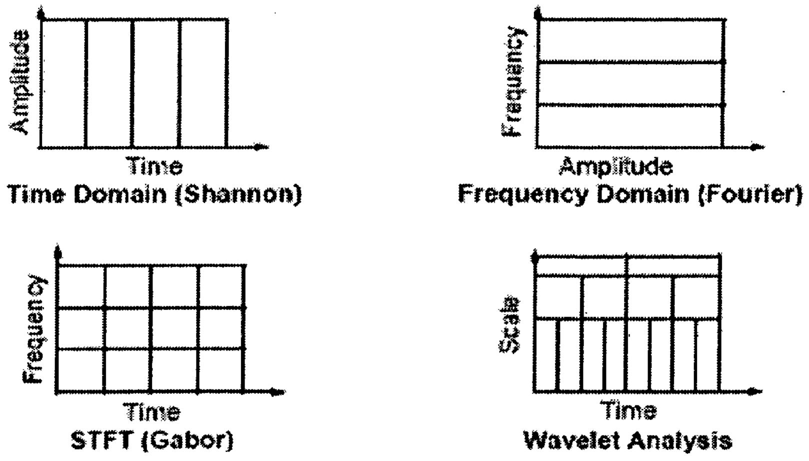
**Wavelet analysis [12]:**

Like Fourier analysis the wavelet transform can be viewed as transforming the signal form the time domain to the wavelet domain [15]. This new domain contains more complicated basis functions called wavelets, mother wavelets or analyzing wavelets.

Wavelet analysis represents the next logical step: a windowing technique with variable-sized regions. Wavelet analysis allows the use of long time intervals where we want more precise low-frequency information, shorter regions where we want high-frequency information.



The following figure shows the difference between all the four methods discussed above those are time domain, Fourier analysis, short time Fourier analysis, and wavelet analysis [14, 15].

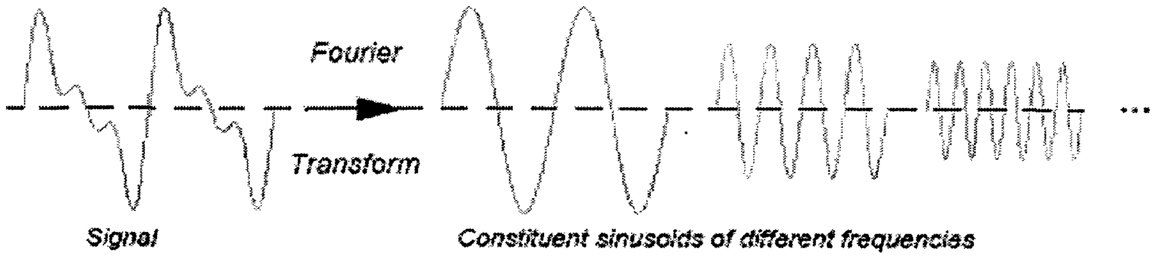


## 5.3 Continuous Wavelet Transform

Mathematically, the process of Fourier analysis is represented by the Fourier transform [14]:

$$F(\omega) = \int_{-\infty}^{\infty} f(t)e^{-j\omega t}dt \qquad \text{..........} \quad 5.1$$

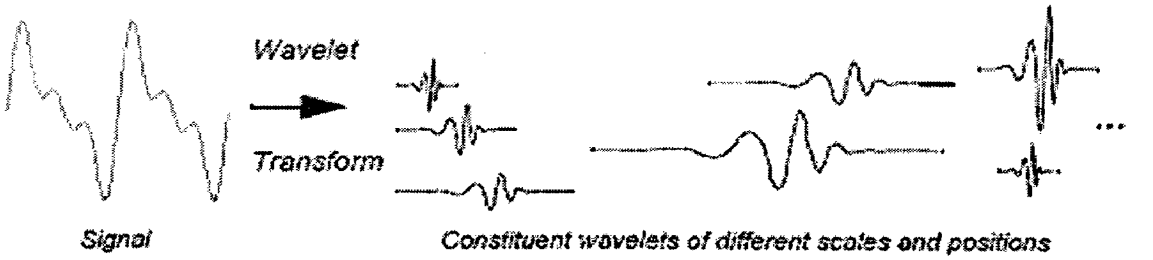
which is the sum over all time of the signal f(t) multiplied by a complex exponential. The results of the transformation are the Fourier coefficients F(ω), which when multiplied by a sinusoid of frequency of ω yield the constituent sinusoidal components of the original signal. Graphically, the process looks like



Similarly, the continuous wavelet transform (CWT) is defined as the sum over all time of the signal multiplied by scaled, shifted versions of the wavelet function Ψ [15]:

$$C(scale, position) = \int_{-\infty}^{\infty} f(t)\psi(scale, position, t)dt \qquad \text{..................}5.2$$

Which is the sum over all time of the signal multiplied by scaled and shifted version of the wavelet function ψ. The results of the CWT are many wavelet coefficients C, which are the function of scale and position. Multiplying each coefficient by the appropriately scaled and shifted wavelet yields the constituent wavelets of the original signal. The basis functions in both Fourier and wavelet analysis are localized in frequency making mathematical tools such as power spectra useful at picking out frequencies and calculating power distributions.



The most important difference between these two kinds of transforms is that individual wavelet functions are localized in space. In contrast Fourier sine and cosine functions are non-local and are active for all time t. This localization feature, along with wavelets localization of frequency, makes many functions and operators using wavelets sparse when transformed it to the wavelet domain. This sparseness, in turn results in a number of useful applications such as data compression, detecting features in images and de-noising signals.

The major draw back of Fourier analysis is that in transforming to the frequency domain, the time domain information is lost. When looking at the Fourier transform of a signal, it is impossible to tell when a particular event took place. In an effect to correct this deficiency, Dennis Gabor (1946) adapted the Fourier transform to analyze only a small section of the signal at a time- a technique called windowing the signal. This is called the Windowed Fourier Transform (WFT). WFT gives information about signals simultaneously in the time domain and in the frequency domain. To illustrate the time-frequency resolution differences between the Fourier transform and the wavelet transform consider the following Figures 5.1 and 5.2 [14].

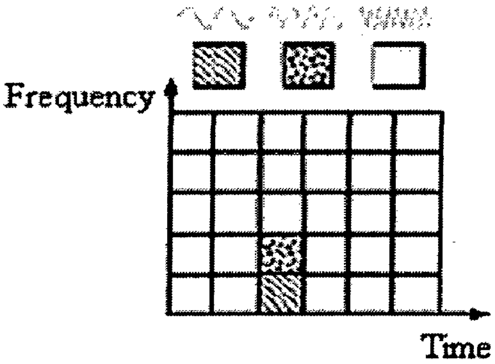


**Figure 5.1 Fourier basis functions and WFT resolution.**

Figure 5.1 shows a windowed Fourier transform, where the window is simply a square wave. The square wave window truncates the sine or cosine function to fit a window of a particular width. Because a signal window is used for all frequencies in the WFT, the resolution of the analysis is the same at all locations in the time frequency plane. An advantage of wavelet transform is that the windows vary. Wavelet analysis allows the use of long time intervals where we want more precise low-frequency information, and shorter regions where we want high frequency information. A way to achieve this is to have short high-frequency basis functions and long low-frequency ones.

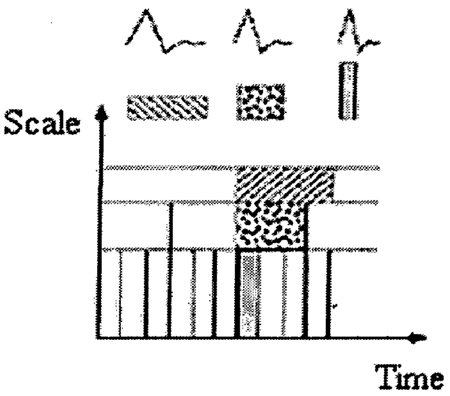


**Figure 5.2 Daubechies wavelet basis functions and wavelets resolution.**

Figure 5.2 shows a time-scale view for wavelet analysis rather than a time frequency region. Scale is inversely related to frequency. A low-scale compressed wavelet with rapidly changing details corresponds to a high frequency. A high scale stretched wavelet that is slowly changing has a low frequency. The figure 5.3 below illustrates four different types of wavelet basis functions.

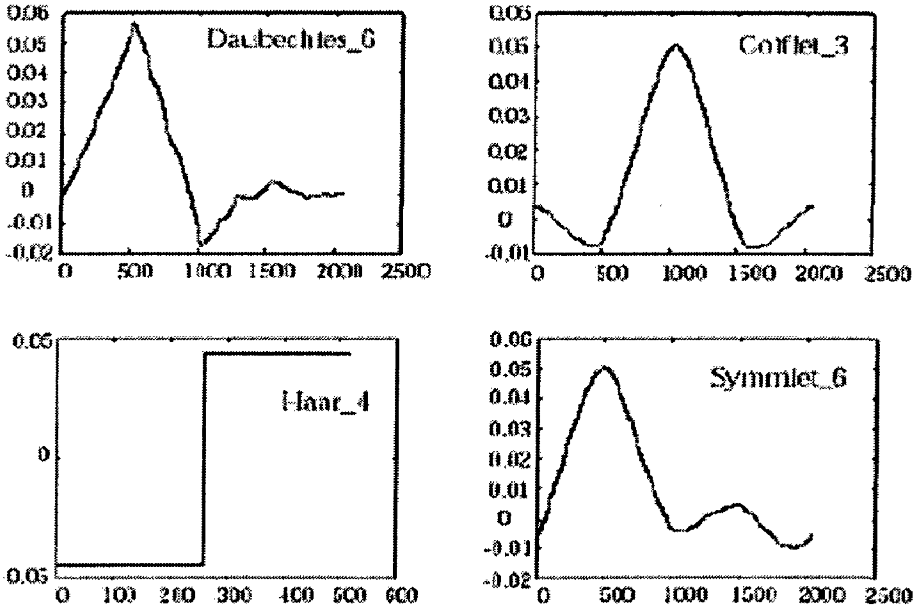


**Figure 5.3 Different Wavelet Families [14]**

The different families make trade-offs between how compactly the basis functions are localized in space and how smooth they are. Within each family of wavelets are wavelet subclasses distinguished by the number of filter coefficients and level of iteration. Wavelets are most often classified with in a family by the number of vanishing moments. This is an extra set of mathematical relationships for the coefficients that must be satisfied. The extent of compactness of signals depends on the number of vanishing moments of the wavelet function used. A more detailed discussion is provided in the next section.

### 5.4 Discrete Wavelet Transform

Calculating wavelet coefficients at every possible scale is a fair amount of work, and it generates lot data. If we chose scales and positions based on powers of two then our analysis will be much more efficient and just as accurate. We obtain such an analysis from the discrete wavelet transform (DWT). The mother wavelet is rescaled or dilated by powers of two and translated by integers. Specifically, a function $f(t) = L^2(R)$ (defines space of square integrable functions) can be represented as

$$f(t) = \sum_{j=1}^{L} \sum_{k=-\infty}^{\infty} d(j,k)\psi(2^{-j}t - k) + \sum_{k=-\infty}^{\infty} a(L,k)\phi(2^{-L}t - k) \quad \text{............5.3}$$

The function $\psi$(t) is known as the mother wavelet, while $\phi$(t) is the scaling function. The set of functions $\{\sqrt{2^{-l}}\phi(2^{-L}t - k), \sqrt{2^{-j}}\psi(2^{-j}t - k) \mid j \le L, j,k,L \in Z\}$, where Z is the set of integers, is an orthonormal basis for $L^2(R)$. The numbers a(L, k) are known as the approximation coefficients at scale L, while d(j, k) are known as the detail coefficients at scale j. the approximation and detail coefficients can be expressed as:

$$a(L,k) = \frac{1}{\sqrt{2^L}} \int_{-\infty}^{\infty} f(t)\phi(2^{-L}t - k)dt \quad \text{......................5.4}$$

$$d(j,k) = \frac{1}{\sqrt{2^j}} \int_{-\infty}^{\infty} f(t)\psi(2^{-j}t - k)dt \quad \text{......................5.5}$$

To provide some understanding of the above coefficients consider a projection $f_l(t)$ of the function f(t) that provides the best approximation (in the sense of minimum error energy) to f(t) at a scale *l*. This projection can be constructed form the coefficients a(L, k), using the equation

$$f_l(t) = \sum_{k=-\infty}^{\infty} a(l,k)\phi(2^{-l}t - k) \quad \text{........................5.6}$$

As the scale *l* decreases, the approximation becomes finer, converging to f(t) as *l*→0. The difference between the approximation at scale *l* + 1 and that at *l*, $f_{l+1}(t) - f_l(t)$, is completely described by the coefficients d(j, k) using the equation

$$f_{l+1}(t) - f_l(t) = \sum_{k=-\infty}^{\infty} d(l,k)\psi(2^{-l}t - k) \quad \text{.................... 5.7}$$

Using these relations, given a(L, k) and {d(j, k) | j ≤ L}, it is clear that we can build the approximation at any scale. Hence the wavelet transform breaks the signal up into a coarse approximation $f_L(t)$. As each layer of detail is added, the approximation at the next finer scale is achieved.

### 5.4.1 Vanishing Moments

The number of vanishing moments of a wavelet indicates the smoothness of the wavelet function as well as the flatness of the frequency response of the wavelet filters. Typically a wavelet with p vanishing moments satisfies the following equation.

$$\int_{-\infty}^{\infty} t^m \psi(t)dt = 0 \qquad \text{for m=0,......, p-1,} \qquad \text{.............. 5.8}$$

or equivalently,

$$\sum_k (-1)^k k^m c(k) = 0 \quad \text{for m=0, ......, p-1.} \qquad \text{......... ..... 5.9}$$

For the representation of smooth signals, a higher number of vanishing moments leads to a fast decay rate of wavelet coefficients. Thus, wavelets with a higher number of vanishing moments lead to a more compact signal representation and are hence useful in coding applications. However, in general, the length of the filters increases with the number of vanishing moments and the complexity of computing the DWT coefficients increases with the size of the wavelet filters.

## 5.5 The Fast Wavelet Transform Algorithm

The Discrete Wavelet Transform (DWT) coefficients can be computed by using Mallat's Fast Wavelet Transform algorithm [14]. This algorithm is sometimes referred to as the two-channel sub-band coder and involves filtering the input signal based on the wavelet function used.

To explain the implementation of the Fast Wavelet Transform algorithm consider the following equations:

$$\phi(t) = \sum_k c(k)\phi(2t-k) \qquad \text{.......... 5.10}$$

$$\psi(t) = \sum_k (-1)^k c(1-k)\phi(2t-k) \qquad \text{...........5.11}$$

$$\sum_k c_k c_{k-2m} = 2\delta_{0,m} \qquad \text{...........5.12}$$

The first equation is known as the twin-scale relation (or the dilation equation) and defines the scaling function $\phi$. The next equation expresses the wavelet $\psi$ in terms of the scaling function function $\phi$. The third equation is the condition required for the wavelet to be orthogonal to the scaling function and its translates. The coefficients c(k) or $\{c_0, \ldots\ldots\ c_{2N-1}\}$ in the above equations represent the impulse response coefficients for a low pass filter of length 2N, with a sum of 1 and a norm of $\frac{1}{\sqrt{2}}$. The high pass filter is obtained form the low pass filter using the relationship $g_k = (-1)^k c(1-k)$, where k varies over the range (1-(2N-1)) to 1. the equation (5.10) shows that the scaling function is essentially a low pass filter and is used to define the approximations. The wavelet function defined by the equation (5.11) is a high pass filter and defines the details. Starting with a discrete input signal vector s, the first stage of the FWT algorithm decomposes the signal into two sets of coefficients. These are the approximation coefficients $cA_1$ (low frequency information) and the detail coefficients $cD_1$ (high frequency information), as shown in the figure below.

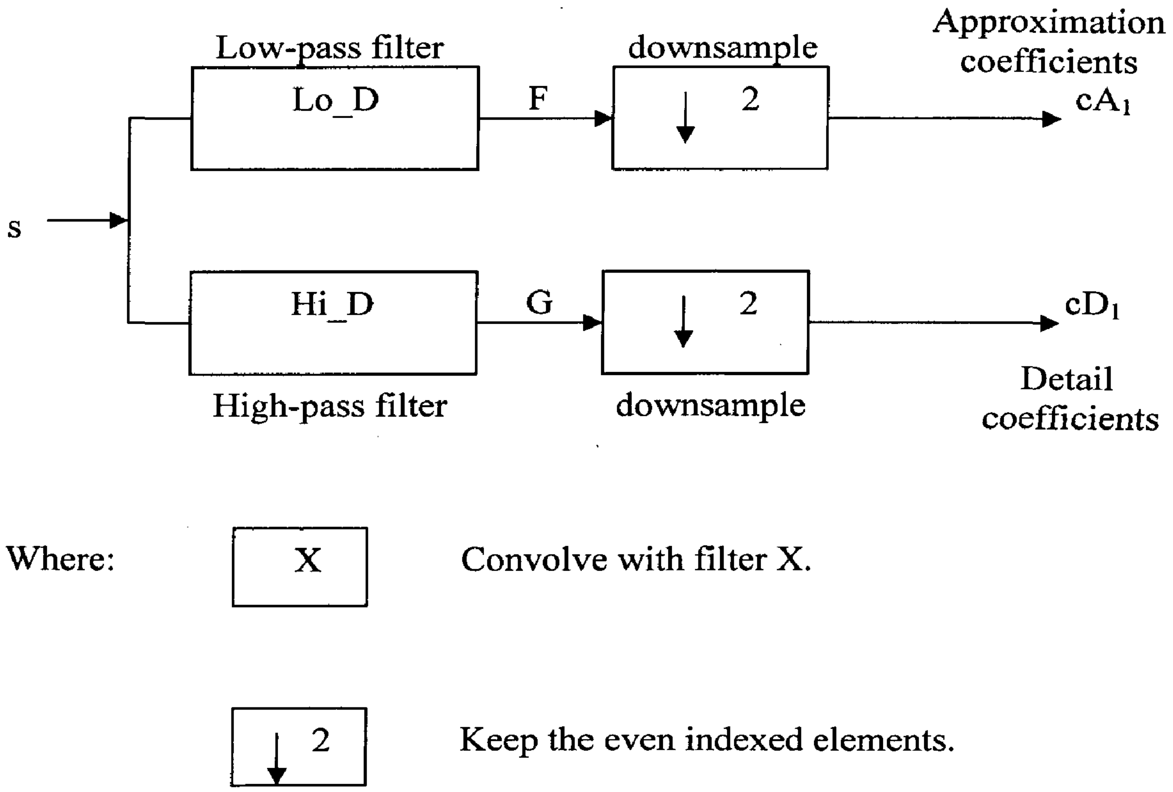


**Figure 5.4 Filtering operation of DWT**

The coefficient vectors are obtained by convolving s with the low-pass filter Lo_D for approximation and with the high-pass filter Hi_D for details. This filtering operation is then followed by dyadic decimation or down sampling by a factor of 2. Mathematically the two-channel filtering of the discrete signal s is represented by the expressions:

$$cA_1 = \sum_k c_k s_{2t-k} \ , \qquad cD_1 = \sum_k g_k s_{2t-k}$$

These equations implement a convolution plus down sampling by a factor 2 and give the forward fast wavelet transform. If the length of the each filter is equal to 2N and the length of the original signal s is equal to n, then the corresponding lengths of the coefficients of $cA_1$ and $cD_1$ are given by the formula:

$$floor(\frac{n-1}{2}) + N \qquad \text{...........5.13}$$

This shows that the total length of the wavelet coefficients is always slightly greater than the length of the original signal due to the filtering process used.

### 5.5.1 Multilevel Decomposition

The decomposition process can be iterated, with successive approximations being decomposed in turn, so that one signal is broken down into many lower resolution components. This is called the wavelet decomposition tree [15].

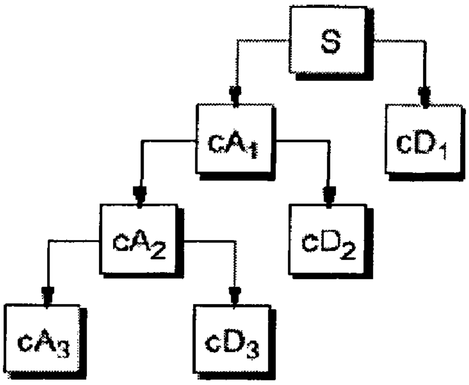


**Figure 5.5 Decomposition of DWT coefficients**

The wavelet decomposition of the signal s analyzed at level j has the following structure [$cA_j$, $cD_j$, ........ , $cD_1$ ]. Looking at a signals wavelet decomposition to level 3 of a sample signal S.

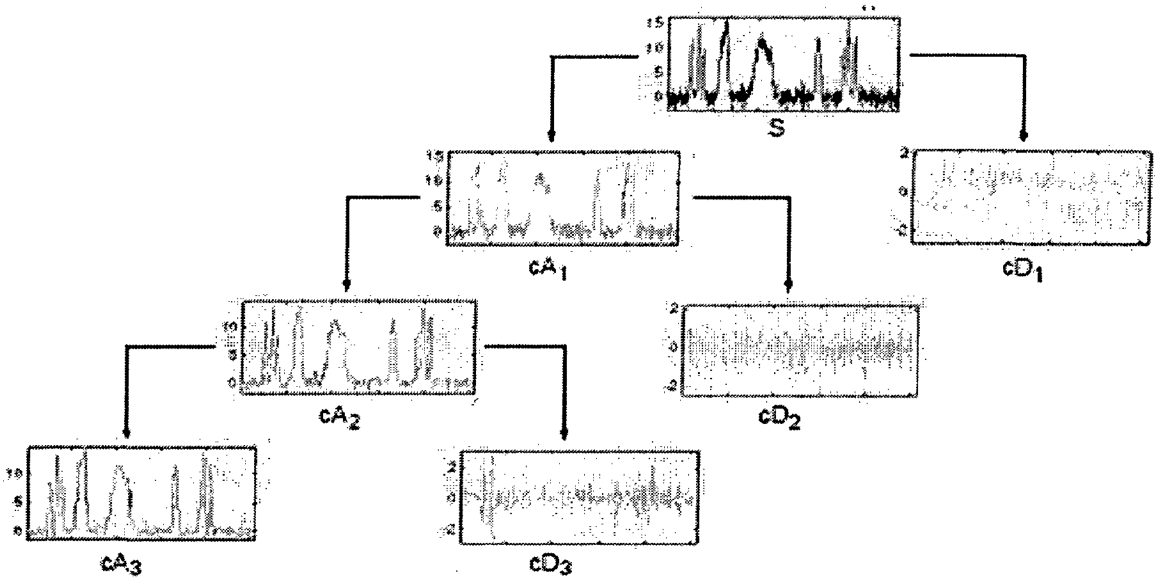


**Figure 5.6 Level 3 decomposition of sample signal**

Since the analysis process is iterative, in theory it can be continued indefinitely. In reality, the decomposition can only proceed until the vector consists of a single sample. Normally, however there is little or no advantage gained in decomposing a signal beyond a certain level. The selection of the optimal decomposition level in the hierarchy depends on the nature of the signal being analyzed or some other suitable criterion, such as low pass filter cut off.

### 5.6 Signal Reconstruction

The original signal can be reconstructed or synthesized using the inverse discrete wavelet transform (IDWT). The synthesis starts with the approximation and detail coefficients $cA_j$ and $cD_j$, and then reconstructs $cA_{j-1}$ by up sampling and filtering with the reconstruction filters.

The reconstruction filters are designed in such a way to cancel out the effects of aliasing introduced in the wavelet decomposition phase. The reconstruction filter (Lo_R and Hi_R) together with the low and high pass decomposition filters forms a system known as quadrature mirror filters (QMF).

For a multilevel analysis, the reconstruction process can itself be iterated producing successive approximations at finer resolutions and finally synthesizing the original signal.

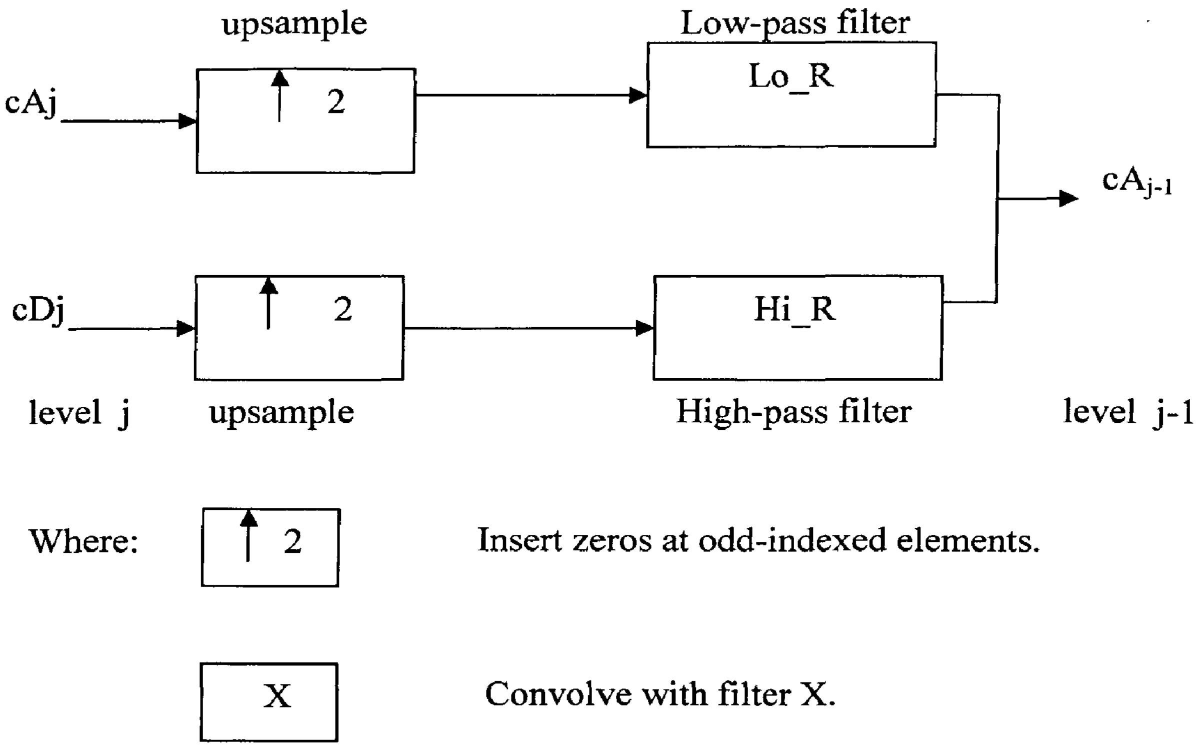


**Figure 5.7 Wavelets Reconstruction**

For many signals, the low-frequency content is the most important part. The high frequency component on the other hand imparts noise. Consider the human voice. If we remove the high frequency components, the voice sounds different, but we can still tell what's being said. However, if we remove the low frequency components, you hear gibberish. In wavelet analysis, we often speak of approximations and details. The approximations are the high-scale, low frequency components of the signal. The details are the low scale, high frequency components. The original signal passes through two complementary filters and emerges as two signals.

### 5.6.1 Optimal Decomposition Level in Wavelet Transforms

The figure below shows a simple speech signal and approximations of the signal, at five different scales. These approximations are reconstructed form the coarse low frequency coefficients in the wavelet transform vector [20]. Figure 5.9 is showing that by keep on increasing the level of decomposition the energy in the approximation part of the signal is decreasing.

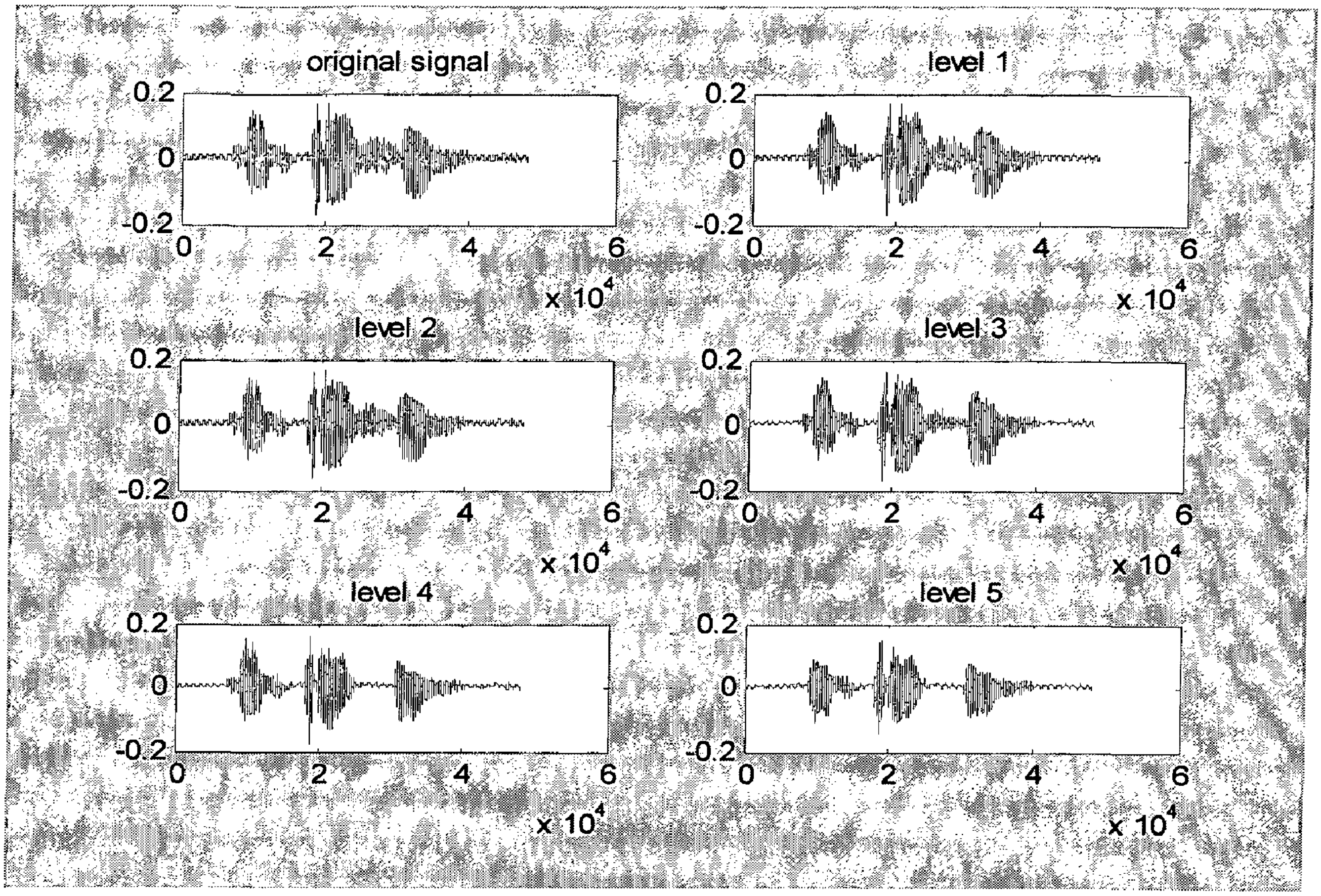


Figure 5.8 Original speech signal and reconstructed approximations

### 5.6.2 Retained Energy in First N/2 Coefficients

A suitable criterion for selecting optimum mother wavelets is related to the amount of energy a wavelet basis function can concentrate into the level 1-approximation coefficients. A speech signal is divided into frames of size 1024 samples and then analyzed using different wavelets. The wavelet transform is computed to scale 5. The signal energy retained in the first N/2 transform coefficients are given in the table given below. This energy is equivalent to the energy stored in the level 1-approximation coefficients [20].

| Wavelet | Avg signal energy retained |
|---|---|
| Haar | 92.57 |
| Db4 | 83.16 |
| Db6 | 96.74 |
| Db8 | 96.81 |
| Db10 | 96.76 |

Table 5.1 Average energy concentrated by different wavelets in N/2 coefficients

## 5.7 Steps of Wavelet Speech Compression

The following Figures 5.9 and 5.10 representing the encoding and decoding blocks of wavelet speech compression.

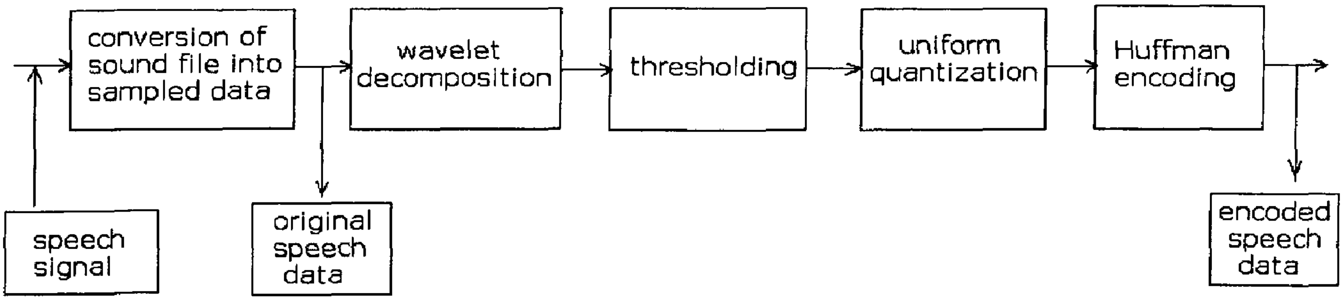


**Figure 5.9 Encoding block diagram of wavelet speech compression**

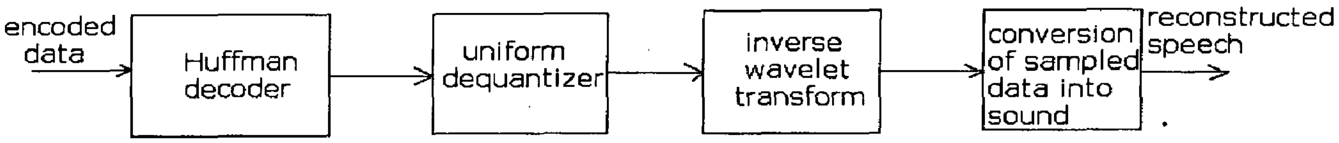


**Figure 5.10 Decoding block diagram of wavelet speech compression**

The process of compressing a speech signal using wavelets involves the following steps [21]:

### 5.7.1 Wavelet Decomposition

The choice of the mother-wavelet function used in designing high quality speech coders is of prime importance. Choosing a wavelet that has compact support in both time and frequency in addition to a significant number of vanishing moments is essential for an optimum wavelet speech compressor [22]. Several different criteria can be used in selecting an optimal wavelet function. The objective is to minimize reconstructed variance and maximize quality. In general optimum wavelets can be selected based on the energy conservation properties in the approximation part of the wavelet coefficients.

In [22] it was shown that the Battle-Lemarie wavelet concentrates more than 97.5% of the signal energy in the approximation part of the coefficients. This is followed very closely by the Daubedhies D20, D12, D10, D8 wavelets, all concentrating more than 96% of the signal energy in the level one approximation coefficients.

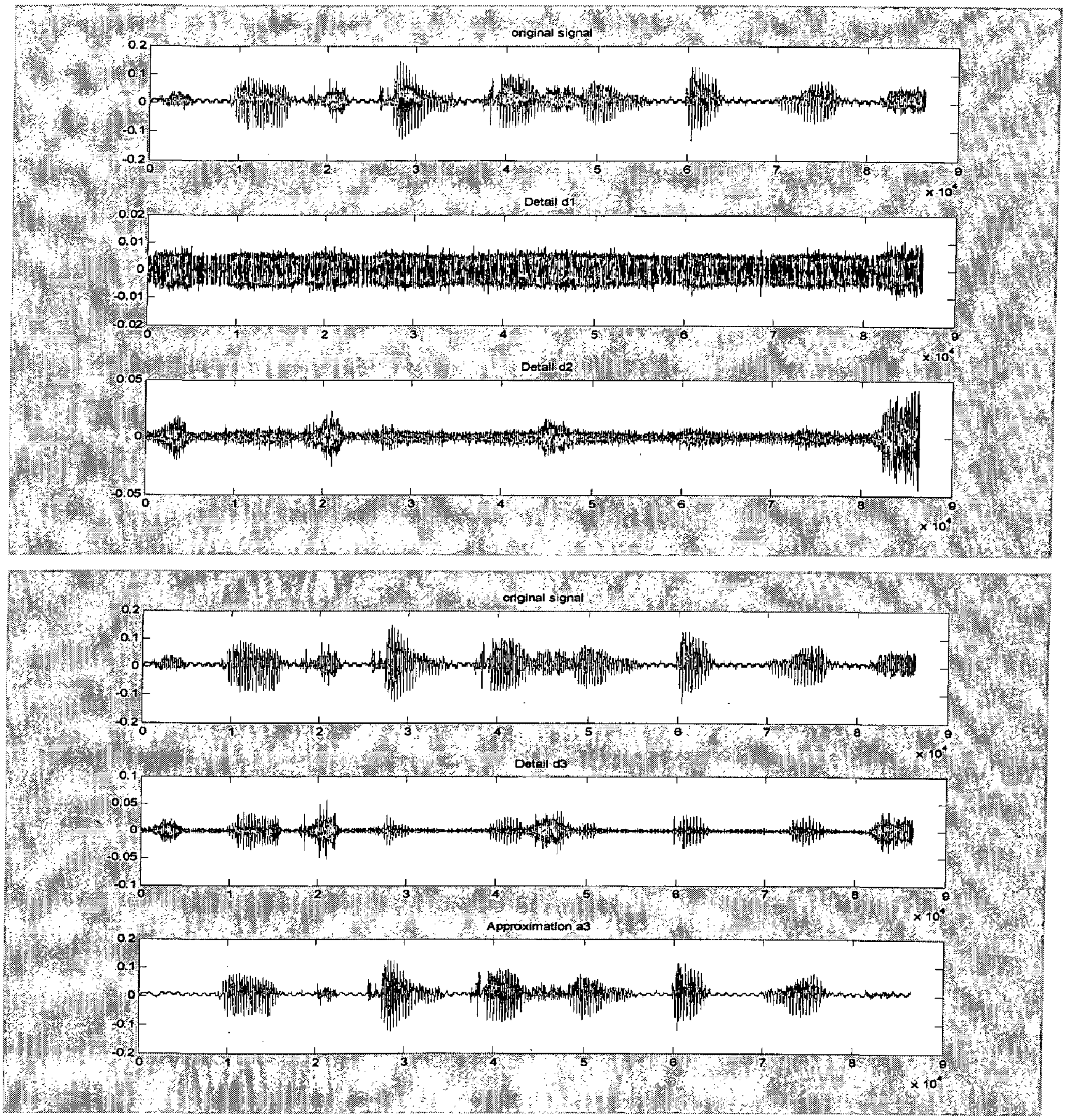


**Figure 5.11 Approximation and detail parts of the speech signal**

Wavelets with more vanishing moments provide better reconstruction quality, as they introduce less distortion into the processed speech and concentrate more signal energy in a few neighboring coefficients. However the computational complexity of the DWT increases with the number of vanishing moments and hence for real time applications it is not practical to use wavelets with an arbitrarily high number of vanishing moments [22]. Wavelets work by decomposing a signal into different resolutions or frequency bands, and this task is carried out by choosing the wavelet function and computing the Discrete Wavelet Transform (DWT). Signal compression is based on the concept that selecting a small number of approximation coefficients (at a suitably chosen level) and some of the detail coefficients can accurately represent regular signal components. Choosing a decomposition level for the DWT usually depends on the type of signal being analyzed or some other suitable criterion such as entropy. For the processing of speech signals decomposition up to scale five is adequate [23], with no further advantage gained in processing beyond scale 5.

### 5.7.2 Threshold

After calculating the wavelet transform of the speech signal, many of the wavelet coefficients are close to or equal to zero. Thresholding can modify the coefficients to produce more zeros. We have two types of thresholding one is level dependent thresholding and another is global thresholding. Level dependent thresholds are calculated using the Brige-Massart strategy [21]. This thresholding scheme is based on an approximation result from Brige and Massart and is well suited for signal compression. This strategy keeps all of the approximation coefficients at the level of decomposition J. The numbers of detail coefficients to be kept at level i starting form 1 to J are given by the formula:

$$n_i = M / (J+2-i)^a$$

a is compression parameter and its value is typically 1.5. The value of M denotes the how scarcely distributed the wavelet coefficients are in the transform vector. If L denotes the length of the coarsest approximation coefficients then M takes on the values in table, depending on the signal being analyzed. For high scarceness M value is L, for medium scarceness M value is 1.5*L and for low M value is 2*L thus this approach to thresholding selects the highest absolute valued coefficients at each level. Where as in the case of global thresholding we select the threshold value in between 0 to Cmax, where Cmax is the maximum coefficient in the decomposition. However this value comes from the final approximation sub signal, and increases the level of decomposition [18].

### 5.7.3 Uniform Quantization

Quantization is a process of mapping a set of continuously valued input data, to a set of discrete valued output data. In other words, the aim of quantization is to decrease the information found in the wavelet coefficients in such a way that this process brings perceptually no error [21]. The process of quantization is shown in the figure 4.4. The use of wavelets and thresholding serves to process the original signal, but to this point, no actual compression of data has yet occurred. This explains that the wavelet analysis does not actually compress a signal, which allows the data to be compressed by standard entropy coding techniques.

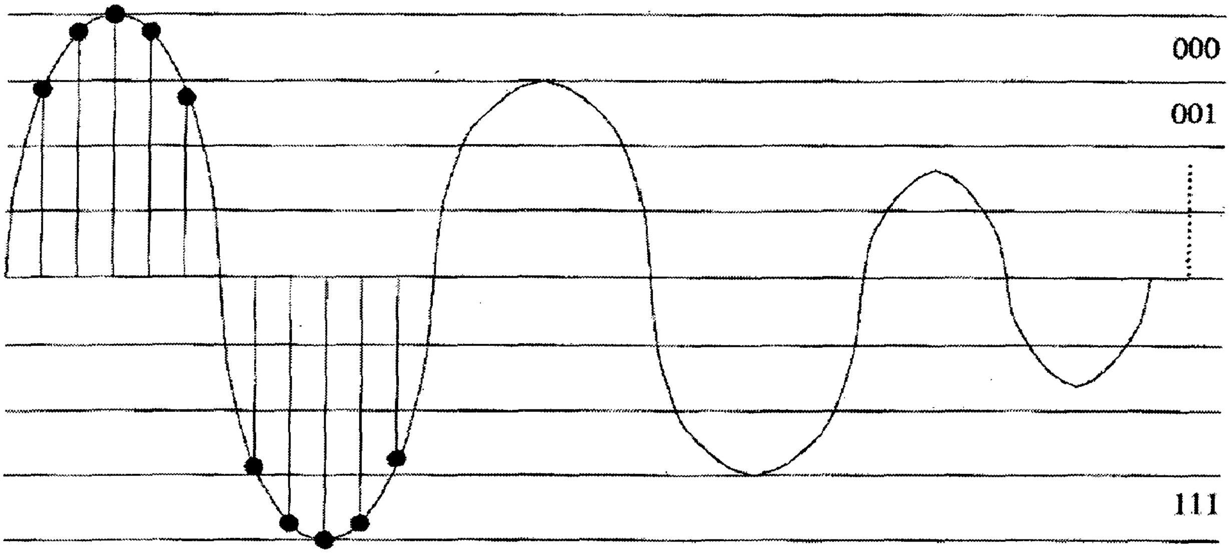


**Figure 5.12 Sampling and quantization of the signal**

The floating point wavelet coefficients are quantized to integer values in this process. These quantized coefficients are the indices to the quantization table. Once the quantization process is done, the quantized value will be fed into the next stage of compression.

### 5.7.4 Huffman Encoding

The quantized data contains redundant information. It is waste of storage space if we were to save the redundancies of the quantized data. One way of overcoming this problem is to use Huffman encoding [23]. In this the probabilities of occurrence of the symbols in the signal are computed. These symbols are the indices to the quantization table. We will sort these symbols according to their probabilities of occurrence in descending order and build the binary tree and codeword table. Due to limitation in the implementation of a binary tree with recursive ability, this encoder uses an array-based binary tree that encodes and decodes the data in a sequence manner. Such an approach incurs expensive computation time. This is the draw back in this coding.

In the real time applications if we want to code any element we need to have minimum of 4 bits for any element in the BCD numbering system. Even though our data is repeating for so many times then also we need to have 4 bits. So if we want to decrease the memory size or the bandwidth of the transmission line we should have to represent the data with less no of bits. This could have been done by using Huffman coding. There are many different reasons for and ways of encoding data and one of these ways is Huffman coding. This is used as a compression method in digital imaging and video as well as in other areas. The idea behind Huffman coding is simply to use shorter bit patterns for more common characters, and longer bit patterns for less common characters. So it is necessary to know the probability of each data element in the data set. Once we know the data and corresponding their probabilities it is possible to encode the data by using Huffman coding.

The steps of Huffman coding [24]:

1. Consider each of the elements as a symbol with its probability.

2. Find the two symbols with the smallest probability and combine them into a new symbol with both letters by adding the probabilities.

3. Repeat step 2 until there is only one symbol left with a probability of 1

4.To see the code, redraw all the symbols in the form of a tree where each symbol contain either a single letter or splits up into two smaller symbols. Label all the left branches of the tree with a 0 and all the right branches with a 1. The code for each of the letters is the sequence of 0's and 1's that lead to it on the tree, starting form the symbol with a probability of 1.

**Example**

If want to encode the letters A(0.12), E(0.42), I(0.09), O(0.30), U(0.07) listed with their respective probabilities. By applying the above steps we can get the following tree.

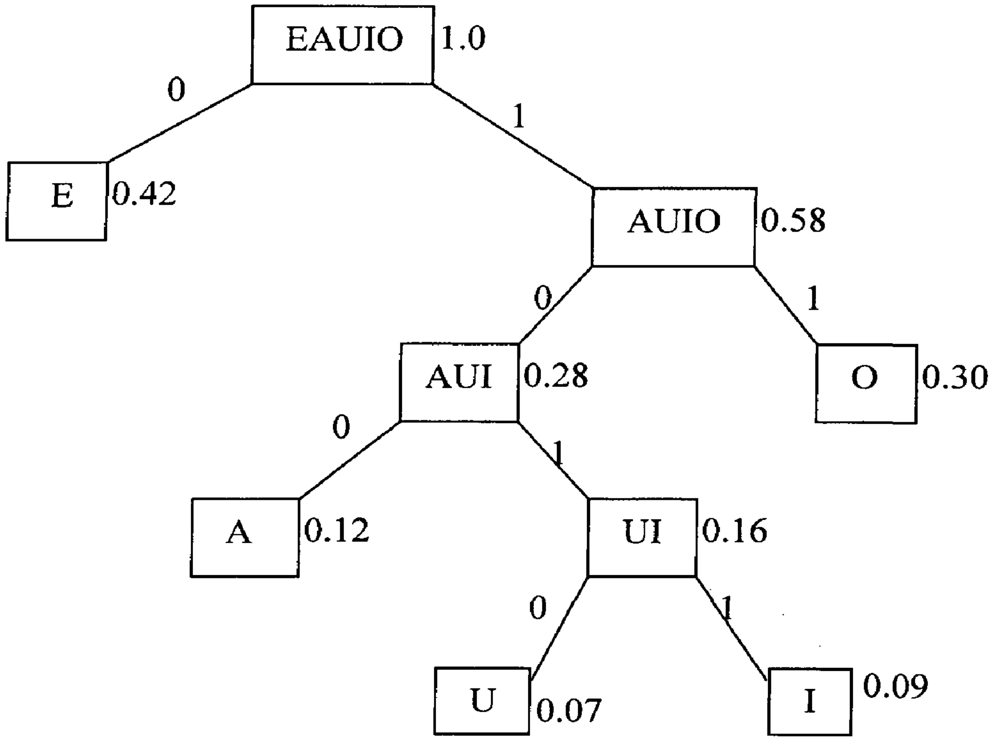


**Figure 5.13 Huffman Coding Tree**

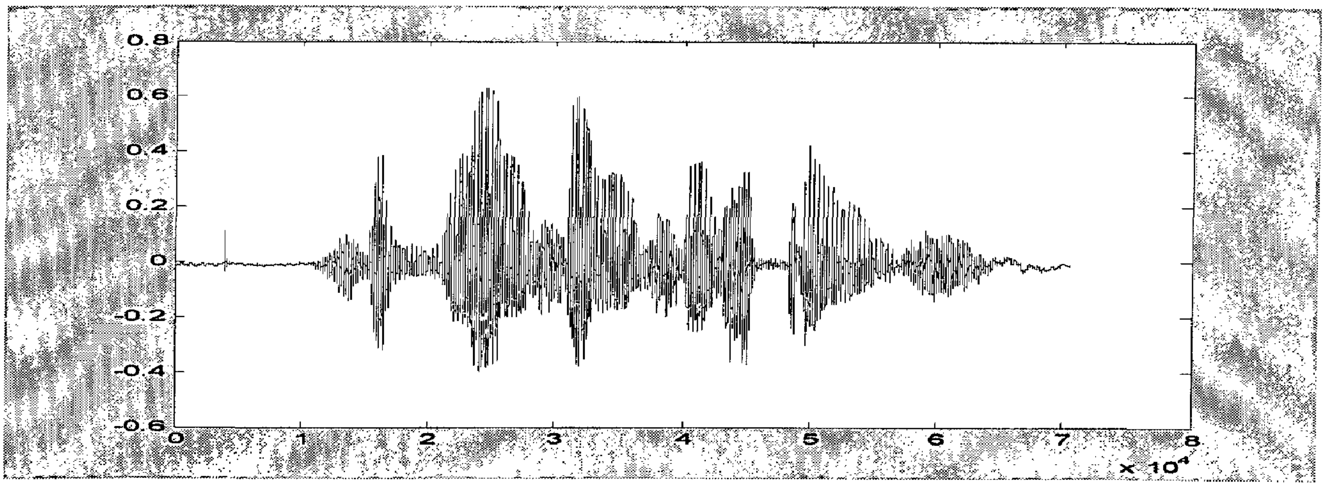
From the above Huffman coding tree we can encode and decode the data. Ex, UEA can be represented as 10100100. Similarly 10110 can be decoded as IE and also any string of vowels can be written uniquely as well as each string of 0's and 1's can be uniquely decoded.

To reconstruct the speech signal, we have to reverse the process for three stages. Those are wavelet transform, quantization and Huffman coding then we can get the reconstructed speech.

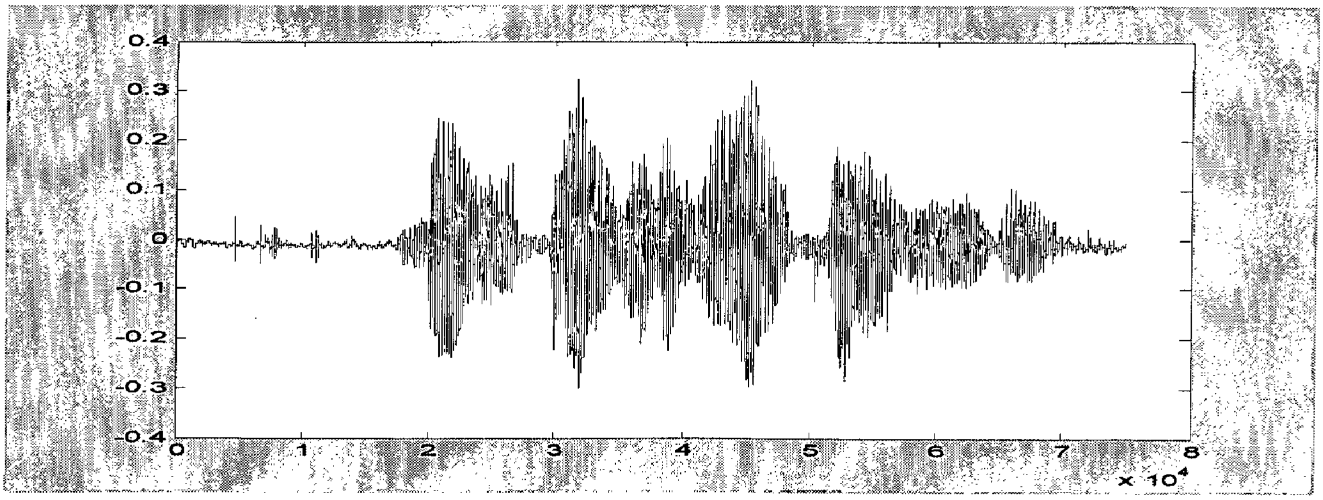
## 6.1 Recording of Speech Patterns

The work is carried out on a Pentium IV PC operating at 2.88 GHz clock with an 8-bit sound card. In this work Matlab 7.0.1 is used for the implementation of speech compression techniques. The speech signals are recorded from software called "jet audio" by using a microphone at a frequency of 44.1 kHz sampled at 16 bits. Signals form five persons has been taken and tested with the three algorithms. Each person had spoken four sentences. All the tested speech signals results are tabulated. Some of the waveforms of recorded speech are shown as below.

The below two waveforms shown in Figure 6.1(a)-(b) represent the same sentence "digital signal processing" that has been spoken by two persons.



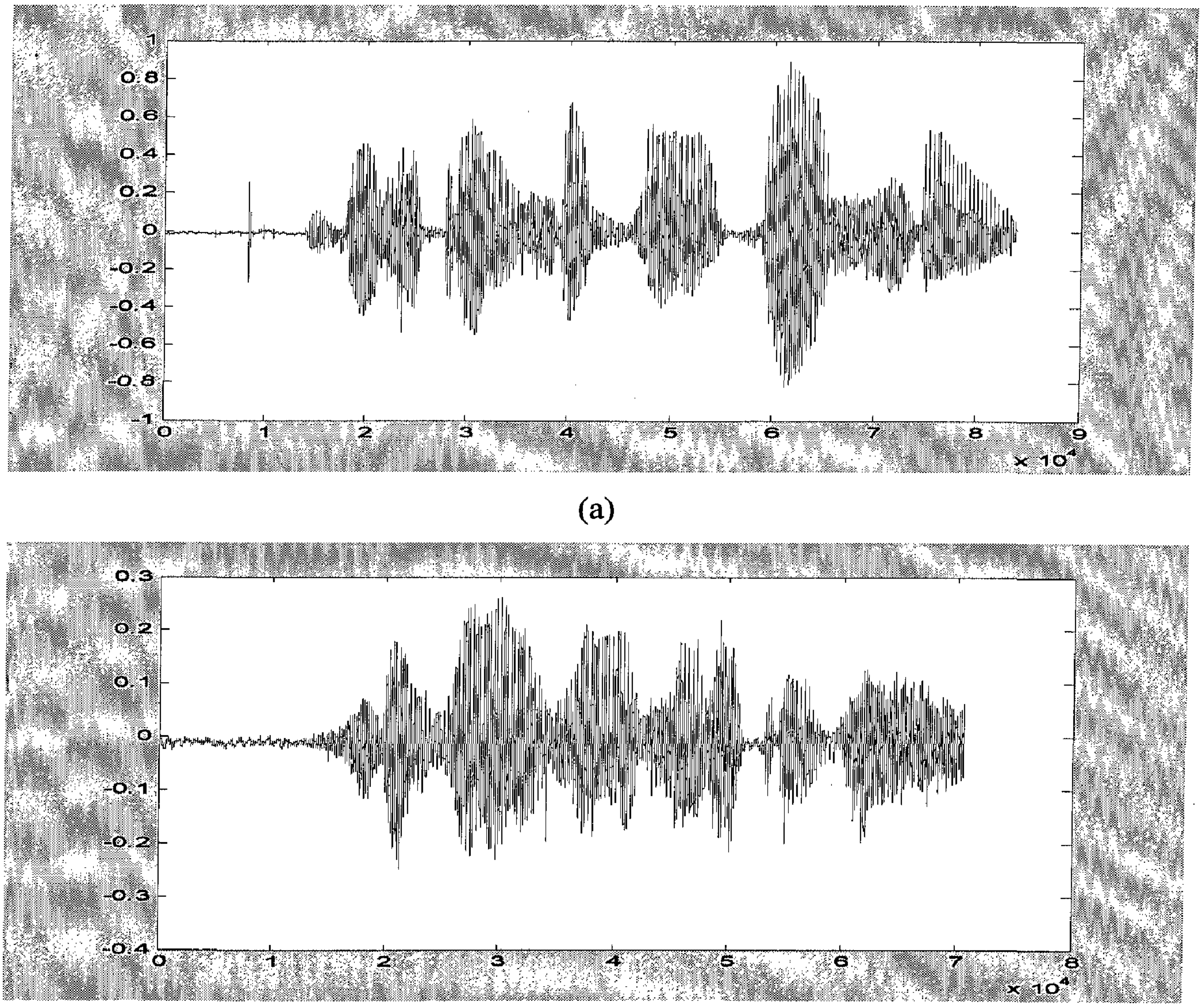
(a)



(b)

**Figure 6.1 Speech waveforms for the sentence "digital signal processing"**

Similarly, the two waveforms shown in Figure 6.2(a)-(b) represent the "signals and systems" spoken by the same persons as before.



(a)

(b)

**Figure 6.2 Speech waveforms for the sentence "signals and systems"**

## 6.2 Parameters for Comparative Evaluation

The following are the parameters for the comparative evaluation of the speech compression techniques that has been implemented in this dissertation work.

1. Compression Ratio
2. Signal to Noise Ratio
3. Peak Signal to Noise Ratio
4. Normalized Root Mean Square Error

The above quantities are calculated using the following formulae:

1. Compression Ratio

$$C = \frac{\text{Memory required for sampled original data}}{\text{Memory required for encoded data}}$$

2. Signal to Noise Ratio [25]

$$SNR = 10\log_{10}(\frac{\sigma_x^2}{\sigma_e^2})$$

$\sigma_x^2$ is the mean square of the speech signal and $\sigma_e^2$ is the mean square difference between the original and reconstructed signals.

3. Peak Signal to Noise Ratio [25]

$$PSNR = 10\log_{10}\frac{NX^2}{\|x - x'\|}$$

N is the length of the reconstructed signal, X is the maximum absolute square value of the signal $x$ and $\|x - x'\|$ is the sum of the square of the difference between the original and reconstructed signals.

$$\|x - x'\| = \sqrt{\sum_{n=1}^{N}(x(n) - x'(n))^2}$$

4. Normalized Root Mean Square Error [25]

$$NRMSE = \sqrt{\frac{\sum_n (x(n) - x'(n))^2}{\mu_n (x(n) - \mu_x(n))^2}}$$

$x(n)$ is the speech signal, $x'(n)$ is the reconstructed signal, and $\mu_n(n)$ is the mean of the speech signal.

$$\mu_n(n) = \frac{\sum_{n=1}^{N} x(n)}{N}$$

### 6.3 Results for LPC Coding

Results of LPC technique are shown in the following four tables from Table 6.1 to Table 6.4. This has been obtained on five speech patterns of same sentence spoken by five different persons. Speech compression has been calculated by storing the original signal data and encoded speech signal data. Original speech signal data and encoded speech signal data are stored in the '.mat' file. The size of the original and reconstructed speech signal data that has been represented in the result tables are in "kbits".

**Table 6.1 Performance index for speech signal 1 ("signals and systems") using LPC coding**

| Person | Original signal size | Size of encoded data | Compression ratio | SNR | PSNR | NRMSE |
|---|---|---|---|---|---|---|
| 1 | 165 | 12.2 | 13.52 | 0.5085 | 1.2499 | 1.0828 |
| 2 | 173 | 15.1 | 11.456 | 0.0919 | -4.8104 | 2.3732 |
| 3 | 126 | 9.94 | 12.696 | 0.1500 | 7.4165 | 1.2711 |
| 4 | 135 | 10.6 | 12.7358 | 0.1451 | -0.3513 | 1.4416 |
| 5 | 148 | 10.6 | 13.960 | 2.2423 | 11.4749 | 1.2033 |

**Table 6.2 Performance index speech signal 2 ("compression") using LPC coding**

| Person | Original signal size | Size of encoded data | Compression ratio | SNR | PSNR | NRMSE |
|---|---|---|---|---|---|---|
| 1 | 84.4 | 6.84 | 12.34 | 0.1520 | 2.7541 | 1.0701 |
| 2 | 86.6 | 7.72 | 11.217 | 0.0630 | 3.9559 | 1.8940 |
| 3 | 84.5 | 6.79 | 12.444 | 0.0738 | 11.2231 | 1.1988 |
| 4 | 80.2 | 6.80 | 11.794 | 0.0553 | 15.2001 | 1.1502 |
| 5 | 114 | 9.09 | 12.541 | 0.2313 | 20.7295 | 1.0385 |

**Table 6.3 Performance index for speech signal 3 ("pulse code modulation") using LPC coding**

| Person | Original signal size | Size of encoded data | Compression ratio | SNR | PSNR | NRMSE |
|---|---|---|---|---|---|---|
| 1 | 171 | 12.8 | 13.359 | 0.1347 | 8.8457 | 1.0362 |
| 2 | 151 | 12.2 | 12.37 | 0.0647 | 6.9544 | 1.2889 |
| 3 | 123 | 9.06 | 13.576 | 0.1343 | 13.4310 | 1.1513 |
| 4 | 129 | 9.96 | 12.951 | 0.0503 | 0.2796 | 1.5488 |
| 5 | 149 | 11.3 | 13.1858 | 1.1269 | 18.0742 | 1.1455 |

**Table 6.4 Performance index for speech signal 4 ("digital signal processing") using LPC coding**

| Person | Original signal size | Size of encoded data | Compression ratio | SNR | PSNR | NRMSE |
|---|---|---|---|---|---|---|
| 1 | 171 | 12.8 | 13.36 | 0.3016 | 4.8623 | 1.0520 |
| 2 | 162 | 12.8 | 12.656 | 0.0764 | 6.5803 | 1.3573 |
| 3 | 147 | 11.2 | 13.125 | 0.1750 | 8.8471 | 1.1791 |
| 4 | 158 | 12.2 | 12.956 | 0.1665 | 4.0152 | 1.2777 |
| 5 | 142 | 10.6 | 13.396 | 1.1816 | 1.6001 | 1.1755 |

The average speech compression in the case of LPC coding is obtained as 12.7809. Average SNR, PSNR, and NRMSE are 0.2062, 7.1166, and 1.2968, respectively. Small variation is there in compression ratios from person to person. The waveforms shown in Figure 6.3 (a)-(e) are the speech patterns obtained before and after compression of speech. On the basis of observations we can say that LPC coding generates more distortion in the wave pattern i.e. quality is poor in the case of LPC coding. This is also proved by SNR, PSNR, and NRMSE measures.

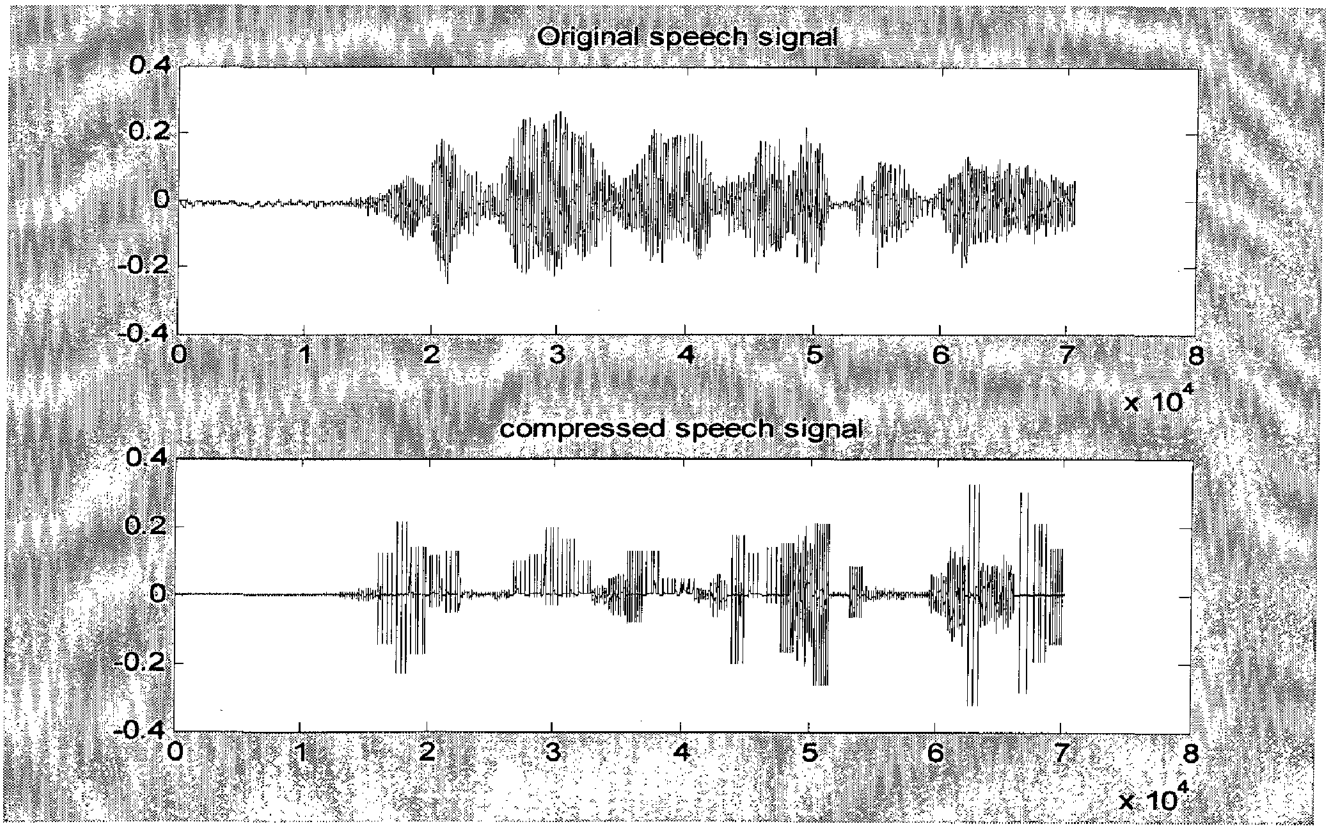


**Figure 6.3(a) Original and reconstructed speech signals ("signals and systems") of person 1 using LPC coding**

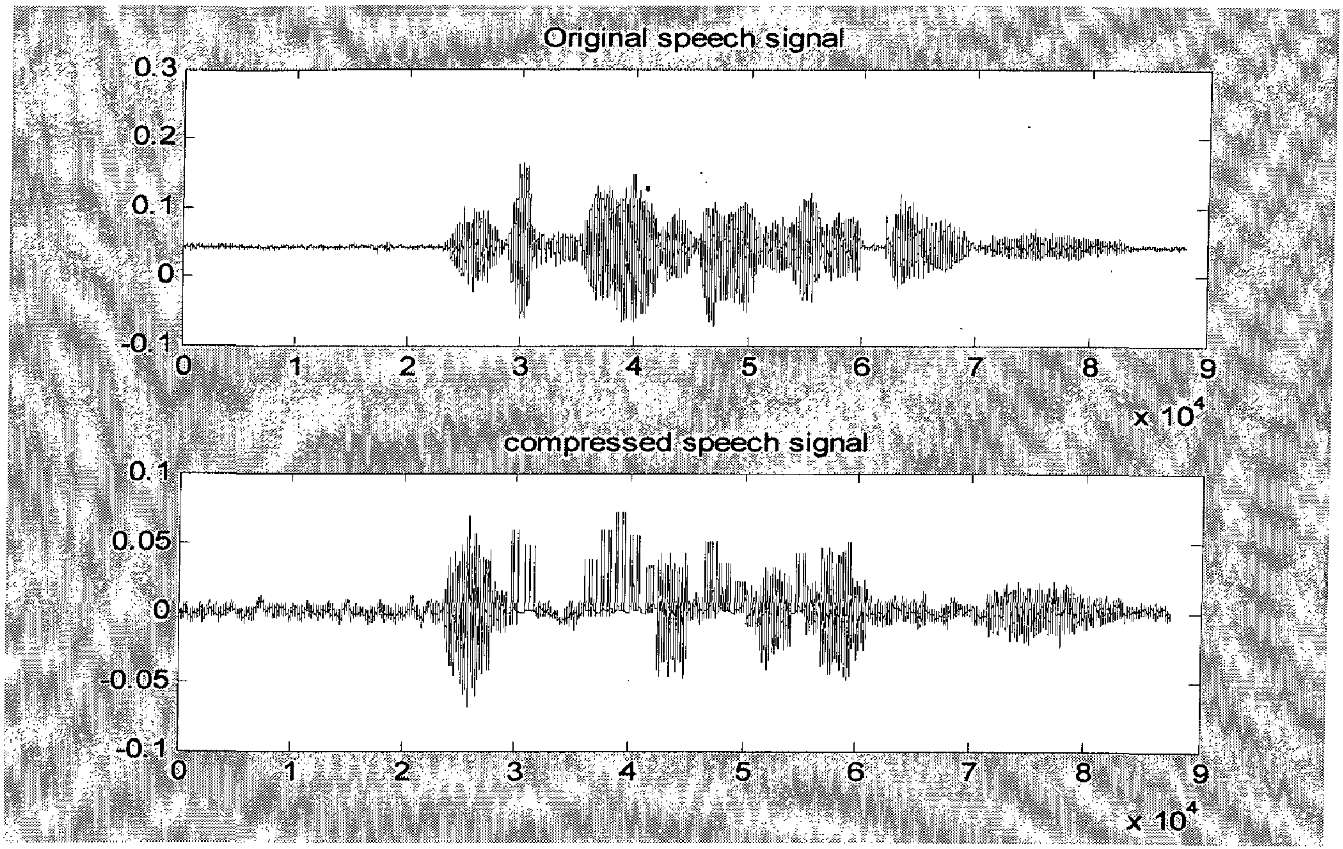


**Figure 6.3(b) Original and reconstructed speech signals ("signals and systems") of person 2 using LPC coding**

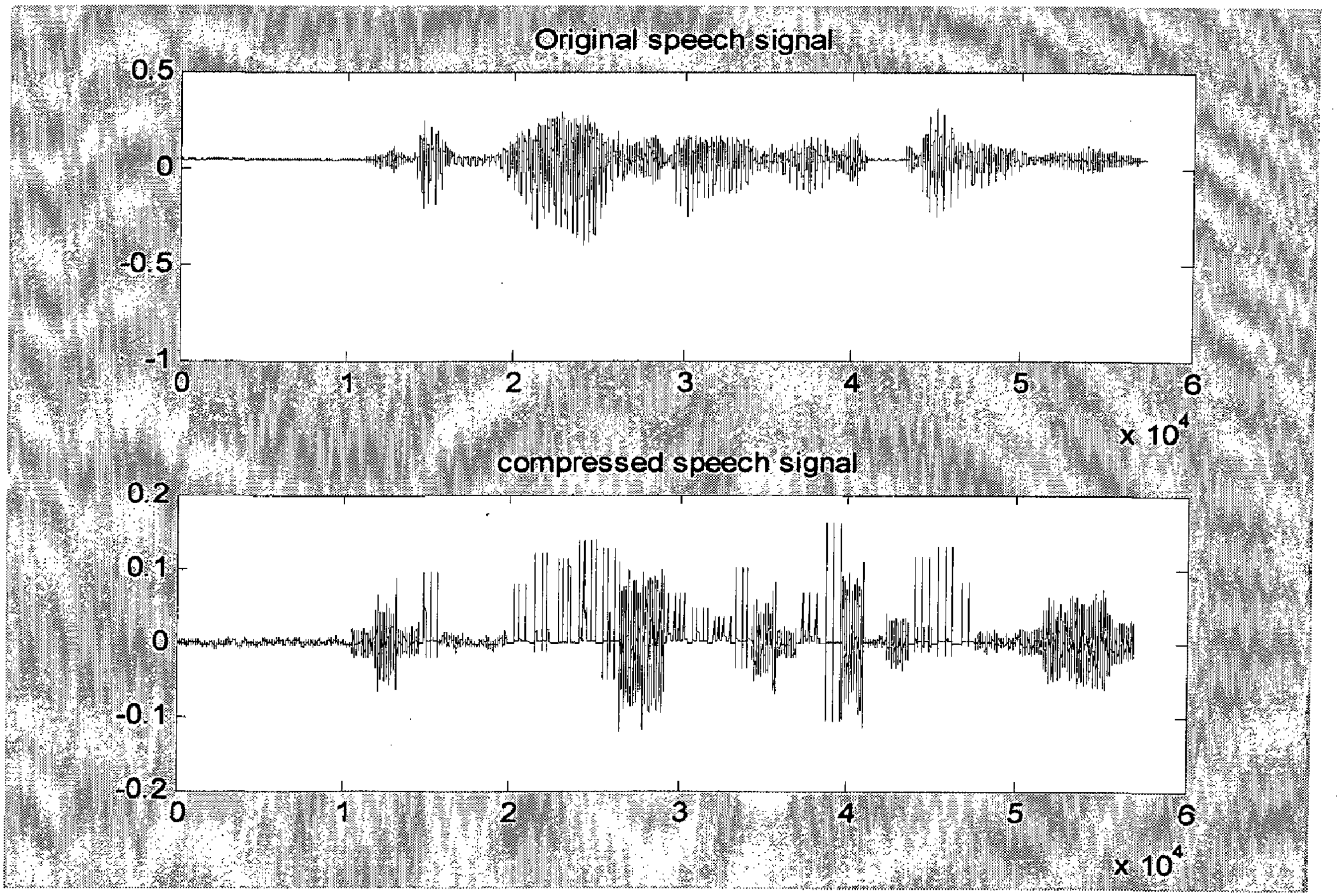


**Figure 6.3(c) Original and reconstructed speech signals ("signals and systems") of person 3 using LPC coding**

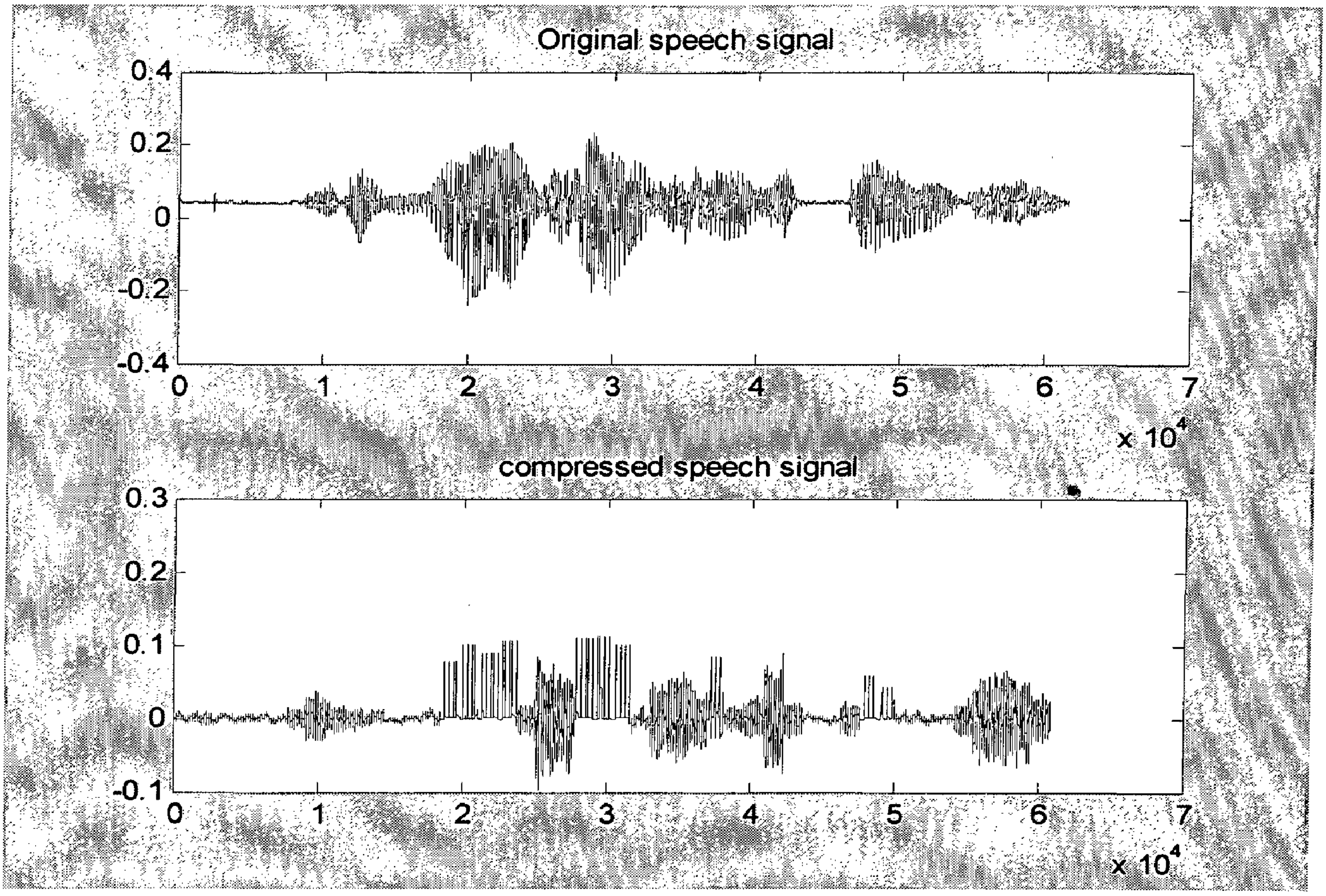


**Figure 6.3(d) Original and reconstructed speech signals ("signals and systems") of person 4 using LPC coding**

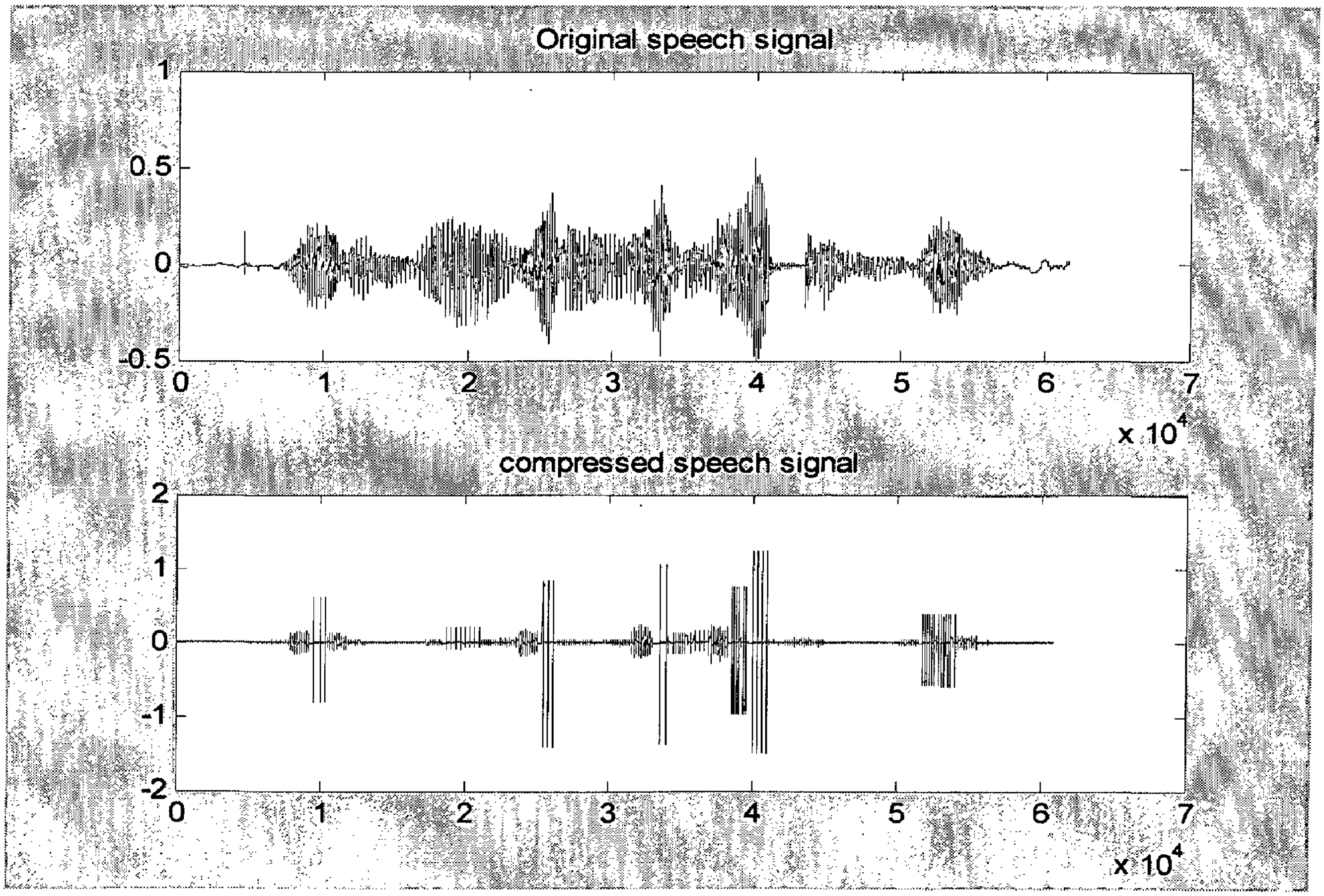


**Figure 6.3(e) Original and reconstructed speech signals ("signals and systems") of person 5 using LPC coding**

### 6.4 Results for ADPCM Technique

Results for ADPCM technique are shown in the following four tables from Table 6.5 to Table 6.6. This has been obtained on four speech signals spoken by five different persons. Speech compression has been calculated by storing the original sampled speech signal data and encoded speech signal data. Original sampled speech signal data and encoded speech signal data are stored in the '.mat' file. The size of the original and decoded speech signal data that has been represented in the result tables are in "kbits".

**Table 6.5 Performance index for speech signal 1 ("signals and systems") using ADPCM technique**

| Person | Original signal size | Size of encoded data | Compression ratio | SNR | PSNR | NRMSE |
|---|---|---|---|---|---|---|
| 1 | 165 | 36.0 | 4.583 | 33.7240 | 29.3780 | 0.0426 |
| 2 | 173 | 45.4 | 3.810 | 32.7744 | 30.5810 | 0.0405 |
| 3 | 126 | 28.9 | 4.359 | 41.9645 | 40.5594 | 0.0281 |
| 4 | 135 | 31.1 | 4.340 | 32.1297 | 30.8879 | 0.0398 |
| 5 | 148 | 31.9 | 4.639 | 26.8942 | 34.2395 | 0.0881 |

**Table 6.6 Performance index for speech signal 2 ("compression") using ADPCM technique**

| Person | Original signal size | Size of encoded data | Compression ratio | SNR | PSNR | NRMSE |
|---|---|---|---|---|---|---|
| 1 | 84.4 | 20.7 | 4.0773 | 28.1990 | 33.4699 | 0.0313 |
| 2 | 86.6 | 22.7 | 3.814 | 33.0659 | 39.6547 | 0.0312 |
| 3 | 84.5 | 20.2 | 4.183 | 39.5300 | 47.6534 | 0.0184 |
| 4 | 80.2 | 20.2 | 3.976 | 34.4465 | 37.2802 | 0.0222 |
| 5 | 114 | 27.6 | 4.130 | 34.6569 | 51.3456 | 0.3010 |

**Table 6.7 Performance index for speech signal 3 ("pulse code modulation") using ADPCM technique**

| Person | Original signal size | Size of encoded data | Compression ratio | SNR | PSNR | NRMSE |
|---|---|---|---|---|---|---|
| 1 | 171 | 38.5 | 4.441 | 33.7860 | 41.6189 | 0.0239 |
| 2 | 151 | 35.4 | 4.265 | 39.9855 | 43.5599 | 0.0191 |
| 3 | 123 | 26 | 4.730 | 34.1032 | 47.1637 | 0.0240 |
| 4 | 129 | 28.1 | 4.590 | 43.1290 | 52.8916 | 0.0151 |
| 5 | 149 | 34.3 | 4.344 | 27.0541 | 42.7272 | 0.0677 |

**Table 6.8 Performance index for speech signal 4 ("digital signal processing") using ADPCM technique**

| Person | Original signal size | Size of encoded data | Compression ratio | SNR | PSNR | NRMSE |
|---|---|---|---|---|---|---|
| 1 | 171 | 38.6 | 4.43 | 29.4003 | 34.2459 | 0.0359 |
| 2 | 162 | 37.8 | 4.285 | 36.0512 | 42.5732 | 0.0216 |
| 3 | 147 | 33.3 | 4.414 | 32.5476 | 42.0508 | 0.0260 |
| 4 | 158 | 35.4 | 4.463 | 56.5396 | 36.6777 | 0.0298 |
| 5 | 142 | 31.8 | 4.465 | 33.4364 | 26.9141 | 0.0642 |

The average speech compression in the case of ADPCM technique is obtained as 4.317. Average SNR, PSNR, and NRMSE are 35.5209, 39.273, and 0.0485, respectively. Small variation is there in compression ratios from person to person. The waveforms shown in Figure 6.4 (a)-(e) are the waveforms that have obtained before and after compression of speech. On the basis of observations form Figure 6.4 we can say that the distortion is less in the case of ADPCM technique, i.e. it gives better quality output. It is also evident by SNR, PSNR, and NRMSE measures.

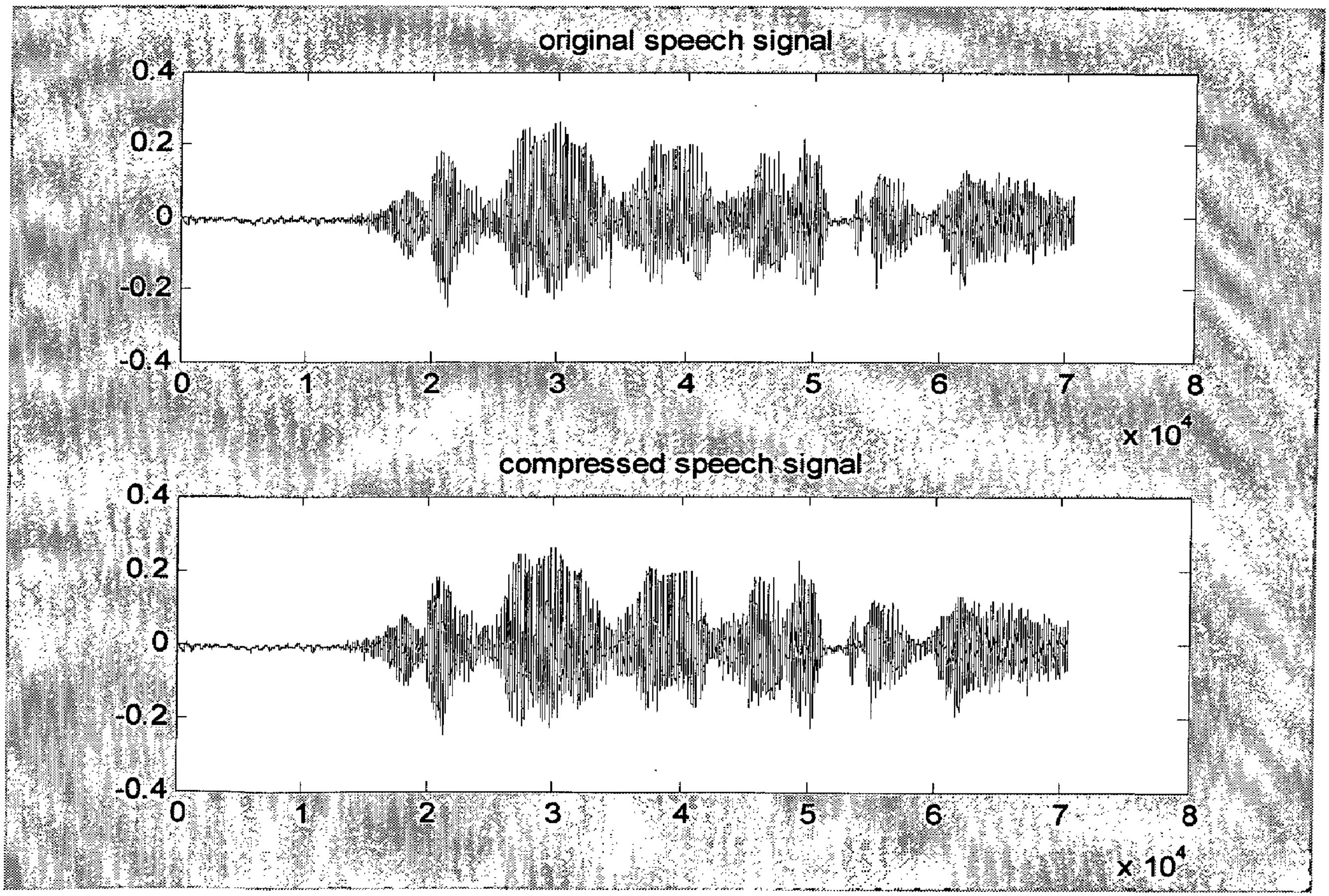


**Figure 6.4(a) Original and reconstructed speech signals ("signals and systems") of person 1 using ADPCM technique**

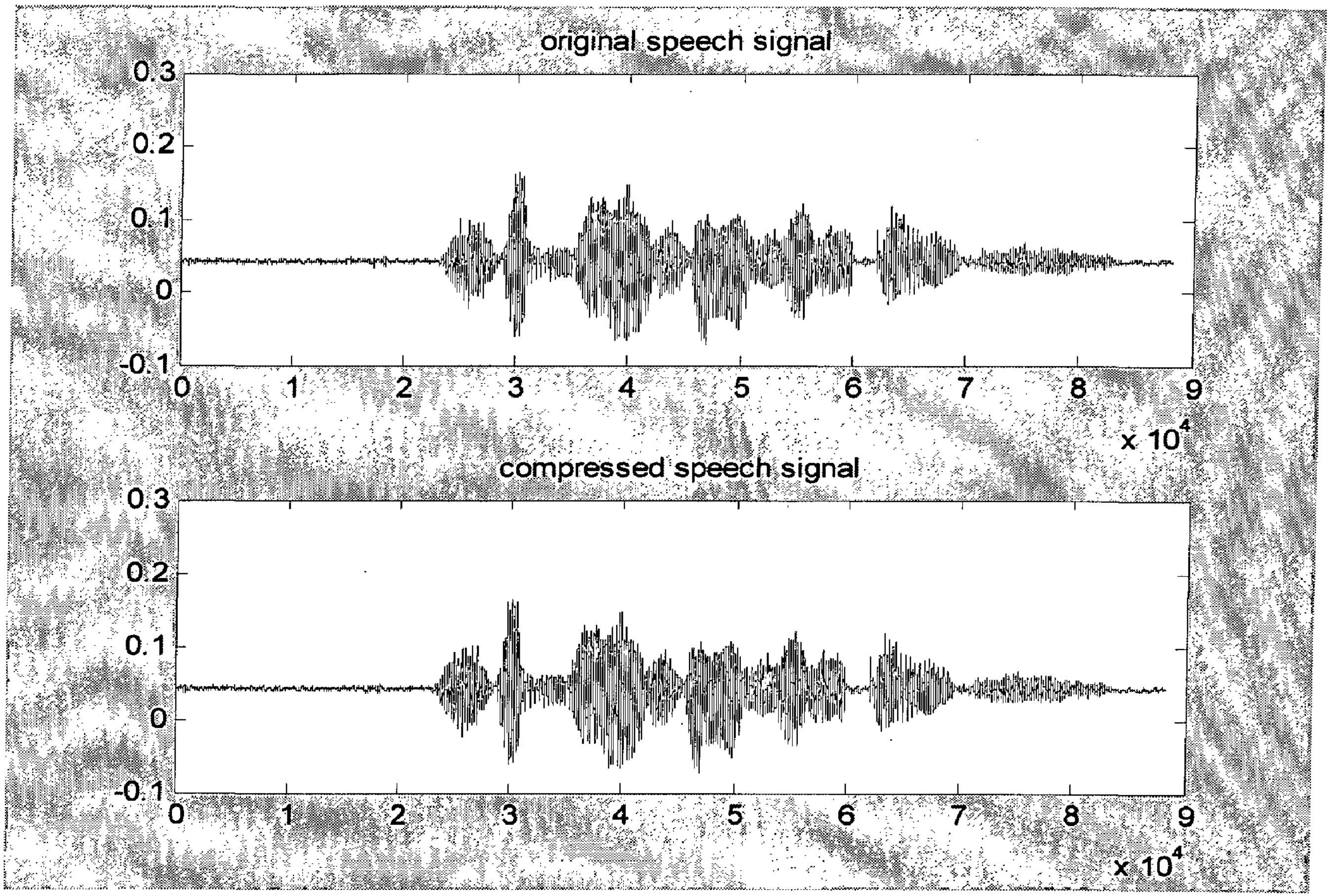


**Figure 6.4(b) Original and reconstructed speech signals ("signals and systems") of person 2 using ADPCM technique**

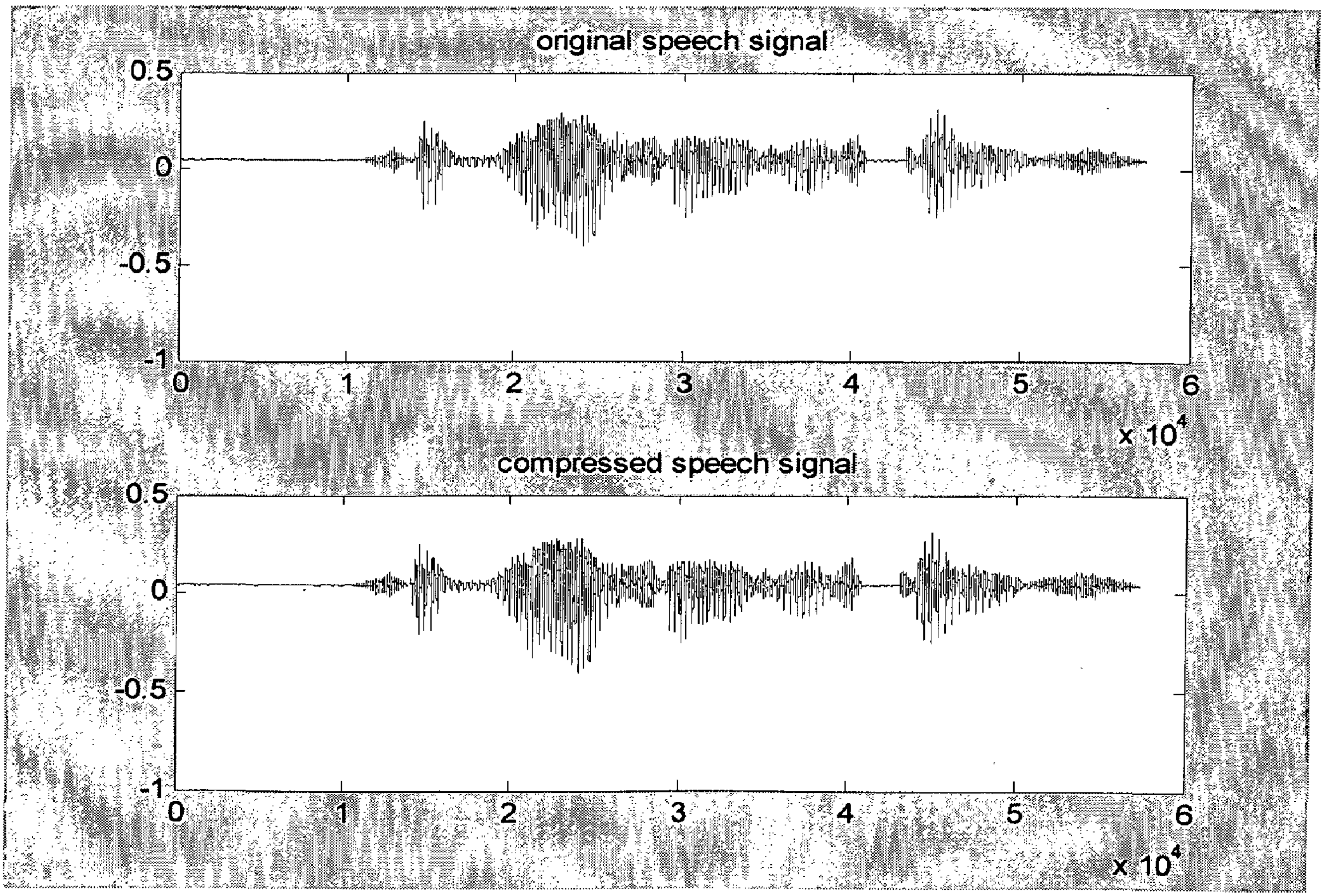


**Figure 6.4(c) Original and reconstructed speech signals ("signals and systems") of person 2 using ADPCM technique**

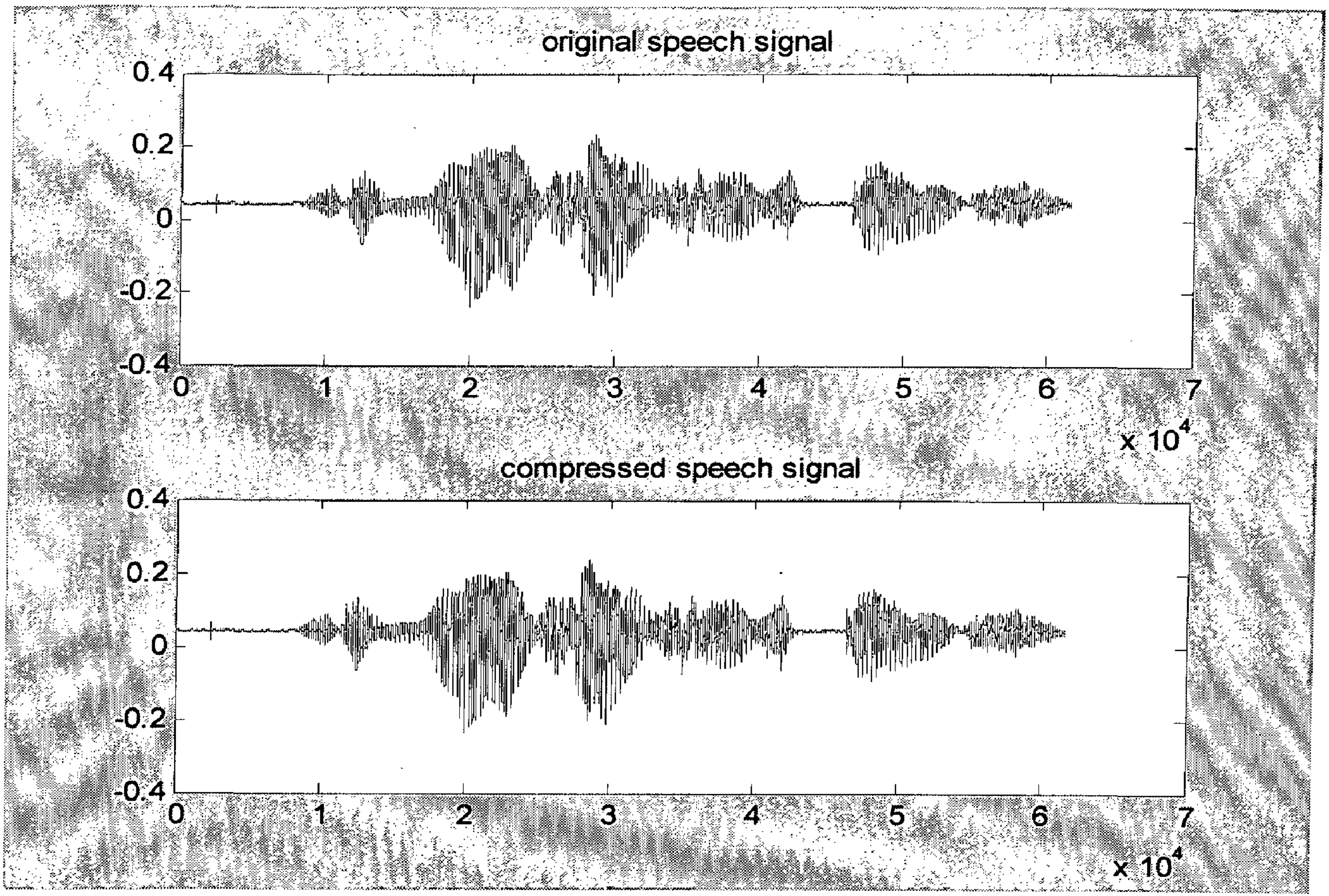


**Figure 6.4(d) Original and reconstructed speech signals ("signals and systems") of person 4 using ADPCM technique**

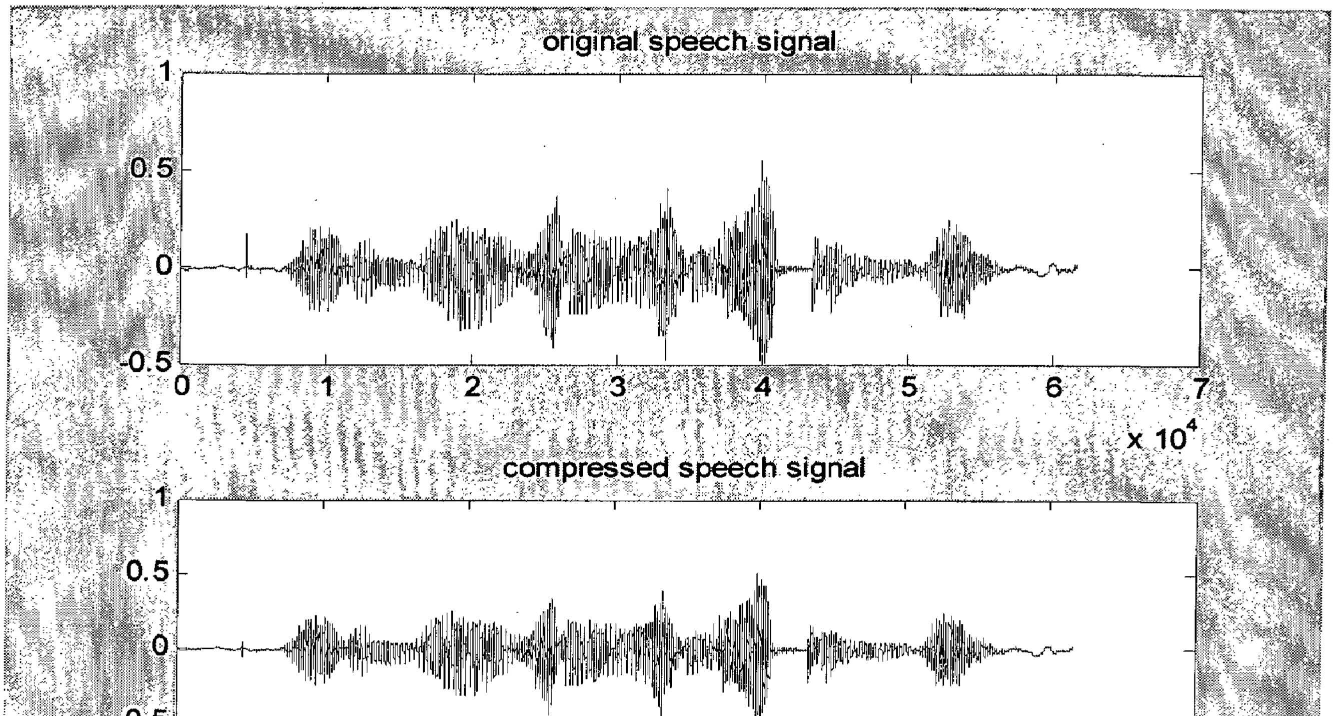


**Figure 6.4(e) Original and reconstructed speech signals ("signals and systems") of person 5 using ADPCM technique**

## 6.6 Results for Wavelet Transform Based Speech Compression.

In wavelet transform based speech compression different wavelets are used like 'Db10', 'Haar', 'Db4', 'Db6', and 'Db8'. The following tables show that the compression ratio is a variable factor and can be varied by varying the decomposition level. A small variation is also experienced in compression ratio form wavelet to wavelet. The following results are for speech signal 1.

**Table 6.9.1 Performance index for speech signal 1 ("compression")**
**(Type of wavelet used: Db10)**

| Level Of decom-position | Original signal size In kbits | Size of encoded data | Compression ratio | SNR | PSNR | NRMSE |
|---|---|---|---|---|---|---|
| 1 | 112 | 16.3 | 6.871 | 12.3464 | 29.9411 | 0.1248 |
| 2 | 112 | 12.4 | 9.032 | 13.7777 | 28.7322 | 0.0906 |
| 3 | 112 | 9.85 | 11.37 | 15.1273 | 31.3083 | 0.0673 |
| 5 | 112 | 5.39 | 20.779 | 13.4857 | 22.5885 | 0.1835 |
| 7 | 112 | 3.00 | 37.333 | 7.6968 | 14.9579 | 0.4418 |

**Table 6.9.2 Performance index for speech signal 1 ("compression")**
**(Type of wavelet used: Haar)**

| Level Of decom-position | Original signal size In kbits | Size of encoded data | Compression ratio | SNR | PSNR | NRMSE |
|---|---|---|---|---|---|---|
| 1 | 112 | 17 | 6.588 | 12.2345 | 8.9396 | 0.8834 |
| 2 | 112 | 14.1 | 7.943 | 13.7118 | 5.5771 | 1.3010 |
| 3 | 112 | 11.1 | 10.09 | 14.4993 | 4.1660 | 1.5305 |
| 5 | 112 | 5.69 | 19.368 | 11.6896 | 4.8200 | 1.4195 |
| 7 | 112 | 3.02 | 37.086 | 6.488 | 7.8135 | 1.0057 |

**Table 6.9.3 Performance index for speech signal 1 ("compression")**
**(Type of wavelet used: Db4)**

| Level Of decom-position | Original signal size In kbits | Size of encoded data | Compression ratio | SNR | PSNR | NRMSE |
|---|---|---|---|---|---|---|
| 1 | 112 | 16.4 | 6.829 | 12.4107 | 10.6389 | 0.7264 |
| 2 | 112 | 12.8 | 8.75 | 13.7792 | 6.5960 | 1.1570 |
| 3 | 112 | 10 | 11.2 | 14.8908 | 4.5852 | 1.4584 |
| 5 | 112 | 5.52 | 20.289 | 13.6031 | 5.1451 | 1.3673 |
| 7 | 112 | 3.01 | 37.209 | 7.267 | 3.216 | 0.1835 |

**Table 6.9.4 Performance index for speech signal 1 ("compression")**
**(Type of wavelet used: Db6)**

| Level Of decom-position | Original signal size In kbits | Size of encoded data | Compression ratio | SNR | PSNR | NRMSE |
|---|---|---|---|---|---|---|
| 1 | 112 | 16.3 | 6.871 | 12.2830 | 12.9139 | 0.5590 |
| 2 | 112 | 12.6 | 8.888 | 13.8817 | 8.0620 | 0.9773 |
| 3 | 112 | 9.9 | 11.313 | 14.8846 | 5.4309 | 1.3231 |
| 5 | 112 | 5.41 | 20.702 | 13.6261 | 3.6946 | 1.6158 |
| 7 | 112 | 3.01 | 37.209 | 6.738 | 7.8111 | 0.5305 |

**Table 6.9.5 Performance index for speech signal 1 ("compression")**
**(Type of wavelet used: Db8)**

| Level Of decom-position | Original signal size In kbits | Size of encoded data | Compression ratio | SNR | PSNR | NRMSE |
|---|---|---|---|---|---|---|
| 1 | 112 | 16.3 | 6.871 | 12.3216 | 17.5710 | 0.3270 |
| 2 | 112 | 12.4 | 9.032 | 13.8247 | 10.6419 | 0.7262 |
| 3 | 112 | 9.79 | 11.44 | 15.6156 | 7.0355 | 1.0999 |
| 5 | 112 | 5.41 | 20.70 | 13.7065 | 3.8861 | 1.5806 |
| 7 | 112 | 3.02 | 37.086 | 6.8831 | 5.4309 | 1.3231 |

On the basis of observation in Table 6.9.1 to Table 6.9.5 it is evident that the compression ratio increases by increasing the level of decomposition. Further, overall observation emphasizes that Db10 wavelet gives the highest compression ratios while the Haar wavelet produces the least compression ratio. Other wavelets give almost same compression ratio. Quality is also better in the case of Db10 filter.

Similar to speech signal 1 results for speech signal 2 are also summarized in tables 6.10.1 to 6.10.5

**Table 6.10.1 Performance index for speech signal 2 ("compression")**
**(Type of wavelet used: Db10)**

| Level Of decom-position | Original signal size In kbits | Size of encoded data | Compression ratio | SNR | PSNR | NRMSE |
|---|---|---|---|---|---|---|
| 1 | 137 | 18.4 | 7.445 | 13.2758 | 30.9791 | 0.1093 |
| 2 | 137 | 14.2 | 9.6478 | 14.2958 | 33.2706 | 0.0840 |
| 3 | 137 | 11.2 | 12.232 | 15.5044 | 35.6874 | 0.0636 |
| 5 | 137 | 6.2 | 22.069 | 13.7321 | 32.1437 | 0.0961 |
| 7 | 137 | 3.65 | 37.534 | 8.5271 | 21.599 | 0.3220 |

**Table 6.10.2 Performance index for speech signal 2 ("compression") (Type of wavelet used: Haar)**

| Level Of decom-position | Original signal size In kbits | Size of encoded data | Compression ratio | SNR | PSNR | NRMSE |
|---|---|---|---|---|---|---|
| 1 | 137 | 19.3 | 7.098 | 13.2610 | 14.7356 | 0.7096 |
| 2 | 137 | 15.7 | 8.726 | 14.3148 | 10.1060 | 1.2092 |
| 3 | 137 | 12.3 | 11.138 | 15.2233 | 8.2583 | 1.4958 |
| 5 | 137 | 6.69 | 20.478 | 12.6689 | 8.8147 | 1.4030 |
| 7 | 137 | 3.68 | 37.228 | 8.2746 | 11.7809 | 0.9971 |

**Table 6.10.3 Performance index for speech signal 2 ("compression") (Type of wavelet used: Db4)**

| Level Of decom-position | Original signal size In kbits | Size of encoded data | Compression ratio | SNR | PSNR | NRMSE |
|---|---|---|---|---|---|---|
| 1 | 137 | 18.5 | 7.405 | 13.3607 | 16.4578 | 0.5820 |
| 2 | 137 | 14.6 | 9.383 | 14.2655 | 11.7341 | 1.0025 |
| 3 | 137 | 11.3 | 12.123 | 15.4043 | 8.6145 | 1.4357 |
| 5 | 137 | 6.4 | 21.406 | 13.2791 | 7.8129 | 1.5745 |
| 7 | 137 | 3.66 | 37.4316 | 9.3288 | 5.6671 | 1.8983 |

**Table 6.10.4 Performance index for speech signal 2 ("compression") (Type of wavelet used: Db6)**

| Level Of decom-position | Original signal size In kbits | Size of encoded data | Compression ratio | SNR | PSNR | NRMSE |
|---|---|---|---|---|---|---|
| 1 | 137 | 18.4 | 7.445 | 13.2630 | 18.4546 | 0.4624 |
| 2 | 137 | 14.4 | 9.514 | 14.3329 | 13.7028 | 0.7992 |
| 3 | 137 | 11.1 | 12.342 | 15.4213 | 9.9936 | 1.2249 |
| 5 | 137 | 6.41 | 21.373 | 13.2791 | 7.8129 | 1.5745 |
| 7 | 137 | 3.66 | 37.4316 | 8.3542 | 7.0591 | 1.1861 |

**Table 6.10.5 Performance index for speech signal 2 ("compression") (Type of wavelet used: Db8)**

| Level Of decom-position | Original signal size In kbits | Size of encoded data | Compression ratio | SNR | PSNR | NRMSE |
|---|---|---|---|---|---|---|
| 1 | 137 | 18.5 | 7.405 | 13.3233 | 22.7200 | 0.2830 |
| 2 | 137 | 14.6 | 9.383 | 14.2989 | 16.3290 | 0.5907 |
| 3 | 137 | 11.5 | 11.913 | 15.5745 | 12.5015 | 0.9177 |
| 5 | 137 | 6.42 | 21.339 | 13.5835 | 8.0648 | 1.5295 |
| 7 | 137 | 3.67 | 37.3297 | 8.6384 | 7.0597 | 1.9639 |

An observation on Table 6.10.1 to Table 6.10.5 reveals that the compression ratio increases by increasing the level of decomposition. For the speech signal 2 also Db10 wavelet gives the highest compression ratios while Haar wavelet produces the least value of compression. All other wavelets produce approximately same compression ratio. Quality is also better in the case of 'Db10' wavelet.

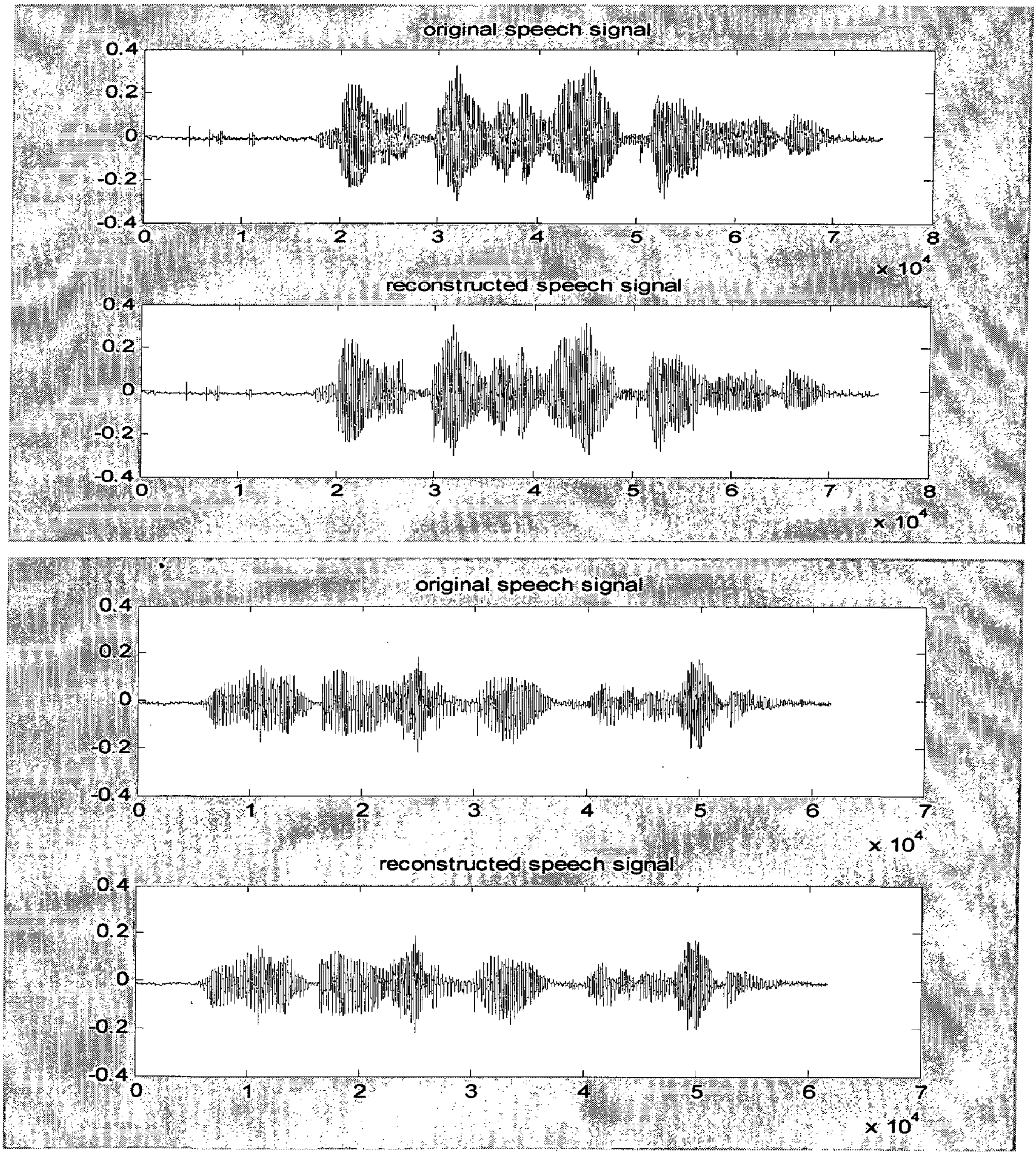


**Figure 6.5 speech signal 1 before and after compression, speech signal 2 before and after compression obtained form Db10 wavelet at decomposition level 3.**

The Figure 6.5 shows the waveforms that have been obtained for wavelet transform based speech compression technique. The waveforms presented are original and reconstructed speech signals at decomposition level 3 from 'Db10'. By observing the Figure 6.5 it is clear that the distortion is less.

The Tables 6.11-6.14 present the summarized results for wavelet transform based speech compression using 'Db10' wavelet at level 3 decomposition for various speech patterns of different persons.

**Table 6.11 Performance index for speech signal 1 ("signals and systems") using WT technique**

| Person | Original signal size | Size of encoded data | Compression ratio | SNR | PSNR | NRMSE |
|---|---|---|---|---|---|---|
| 1 | 165 | 17.1 | 9.649 | 22.4858 | 19.1364 | 0.1385 |
| 2 | 173 | 13.4 | 12.910 | 12.7169 | 19.5538 | 0.1443 |
| 3 | 127 | 13.3 | 9.548 | 15.3050 | 31.1091 | 0.0834 |
| 4 | 135 | 13.7 | 9.854 | 14.4309 | 22.7079 | 0.1021 |
| 5 | 148 | 16.5 | 8.969 | 12.5653 | 24.9766 | 0.2561 |

**Table 6.12 Performance index for speech signal 2 ("compression") using WT technique**

| Person | Original signal size | Size of encoded data | Compression ratio | SNR | PSNR | NRMSE |
|---|---|---|---|---|---|---|
| 1 | 84.4 | 8.09 | 10.432 | 16.1730 | 26.2045 | 0.0724 |
| 2 | 86.6 | 8.10 | 10.691 | 13.4488 | 31.0362 | 0.0842 |
| 3 | 84.5 | 10 | 8.45 | 15.9875 | 39.3480 | 0.0478 |
| 4 | 80.2 | 7.87 | 10.190 | 14.2175 | 27.8808 | 0.0656 |
| 5 | 114 | 12.51 | 9.110 | 15.7230 | 29.2417 | 0.0522 |

**Table 6.13 Performance index for speech signal 3 ("pulse code modulation") using WT technique**

| Person | Original signal size | Size of encoded data | Compression ratio | SNR | PSNR | NRMSE |
|---|---|---|---|---|---|---|
| 1 | 171 | 17.6 | 9.715 | 21.9355 | 34.4231 | 0.0548 |
| 2 | 151 | 15.3 | 9.869 | 15.7762 | 34.8348 | 0.0522 |
| 3 | 123 | 13.0 | 9.411 | 14.8593 | 36.4426 | 0.0714 |
| 4 | 129 | 13.8 | 9.351 | 17.3576 | 28.5211 | 0.0628 |
| 5 | 149 | 16.1 | 9.250 | 26.0456 | 40.4920 | 0.0876 |

**Table 6.14 Performance index for speech signal 4 ("digital signal processing") using WT technique**

| Person | Original signal size | Size of encoded data | Compression ratio | SNR | PSNR | NRMSE |
|---|---|---|---|---|---|---|
| 1 | 171 | 17.8 | 9.609 | 23.5541 | 26.2899 | 0.898 |
| 2 | 162 | 14.5 | 11.160 | 18.4971 | 19.3461 | 0.3254 |
| 3 | 147 | 14.6 | 9.541 | 16.3425 | 16.5732 | 0.1732 |
| 4 | 158 | 16.0 | 9.854 | 17.2574 | 25.9342 | 0.2245 |
| 5 | 142 | 13.8 | 10.289 | 21.5641 | 18.8945 | 0.1617 |

The average speech compression in the case of WT technique is obtained as 9.9587. Average SNR, PSNR, and NRMSE are 17.1575, 28.349, and 0.2424, respectively. Small variation is there in compression ratios from person to person. The waveforms shown in Figure 6.6 (a)-(e) are obtained before and after compression of the speech. On the basis of the observations it can be said that the distortion is less in the case of WT technique at level 3 decomposition using 'db10' wavelet, i.e. it gives good quality output which is also proved by SNR, PSNR, and NRMSE measure.

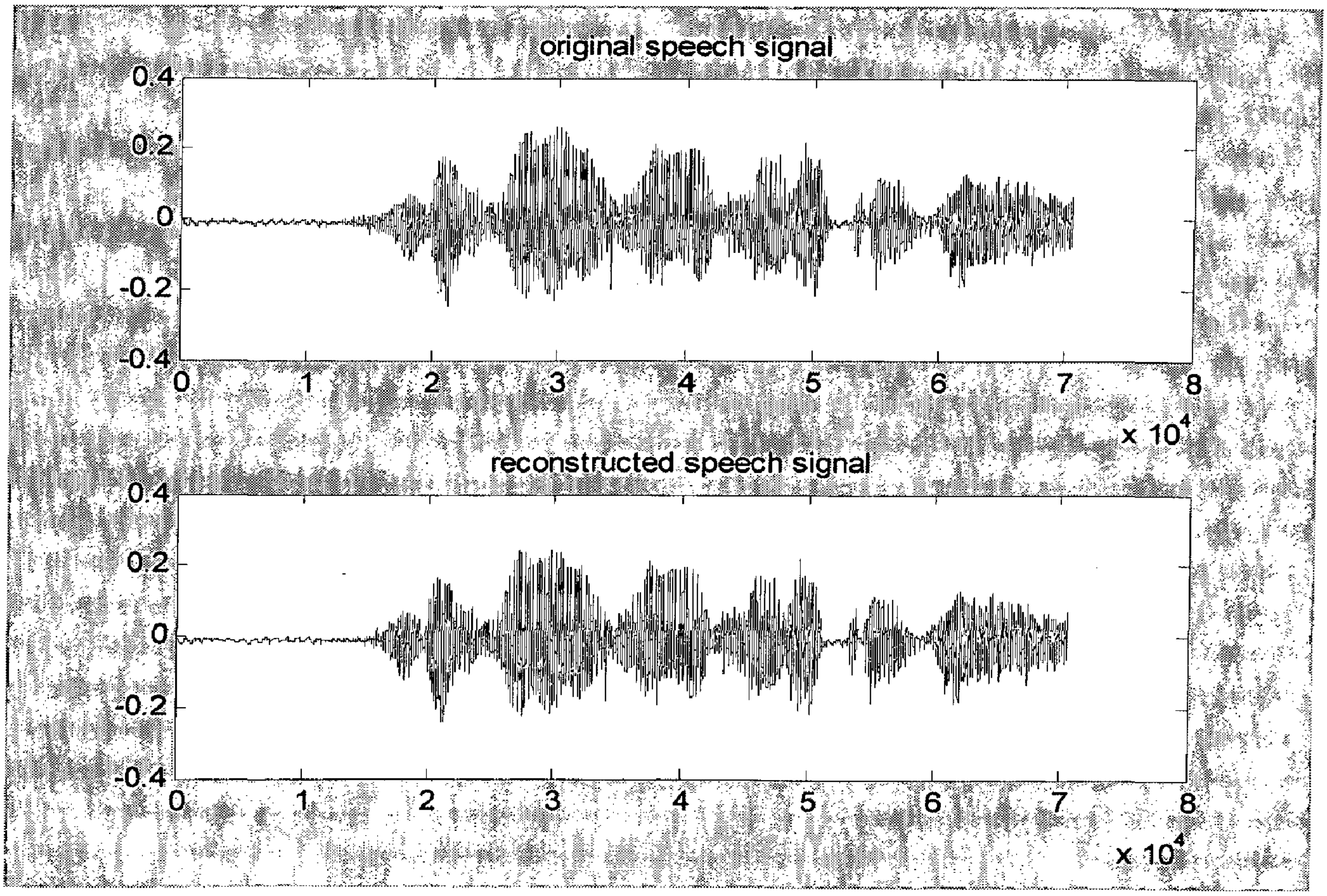


**Figure 6.6(a) Original and reconstructed speech signals ("signals and systems") of person 1 using WT technique**

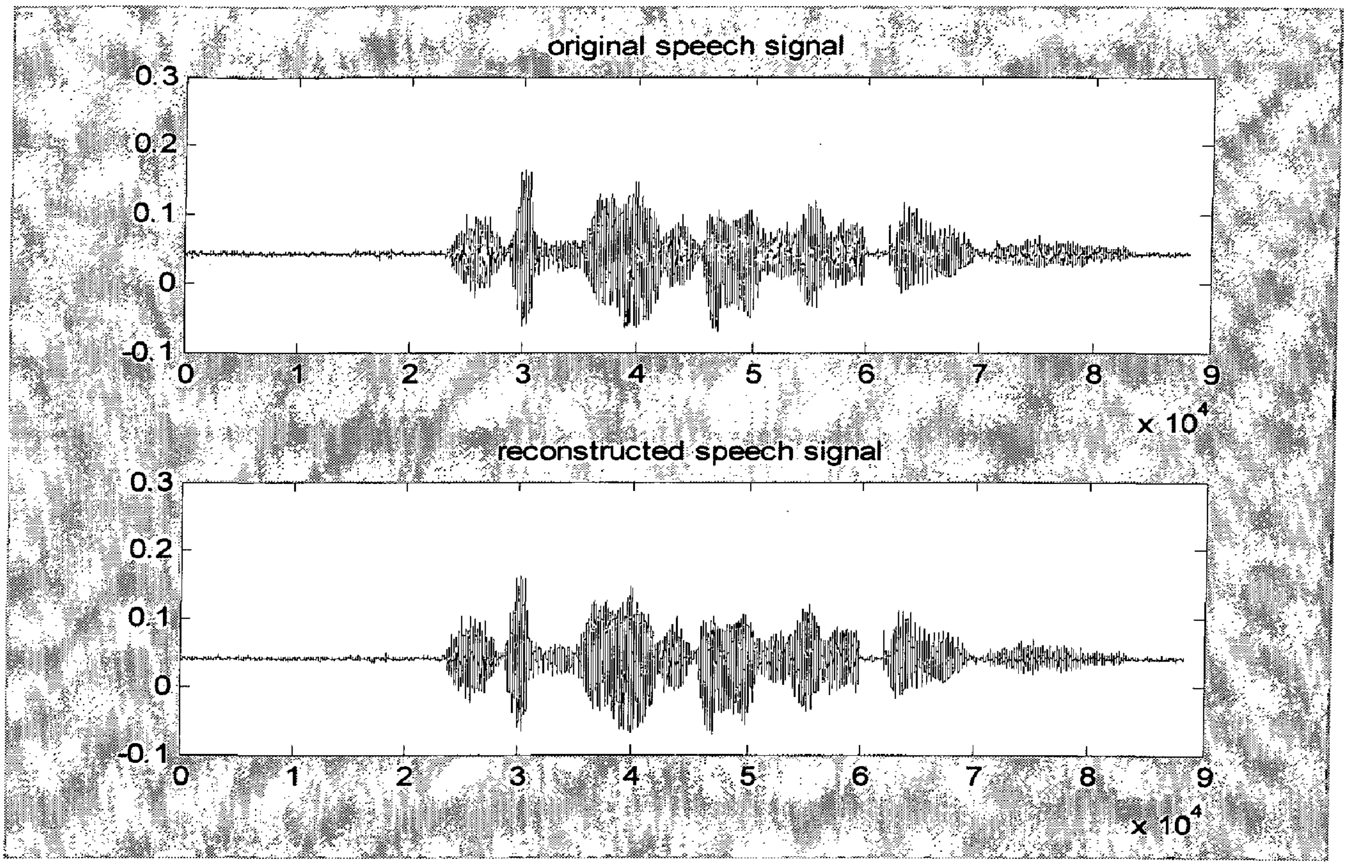


**Figure 6.6(b) Original and reconstructed speech signals ("signals and systems") of person 2 using WT technique**

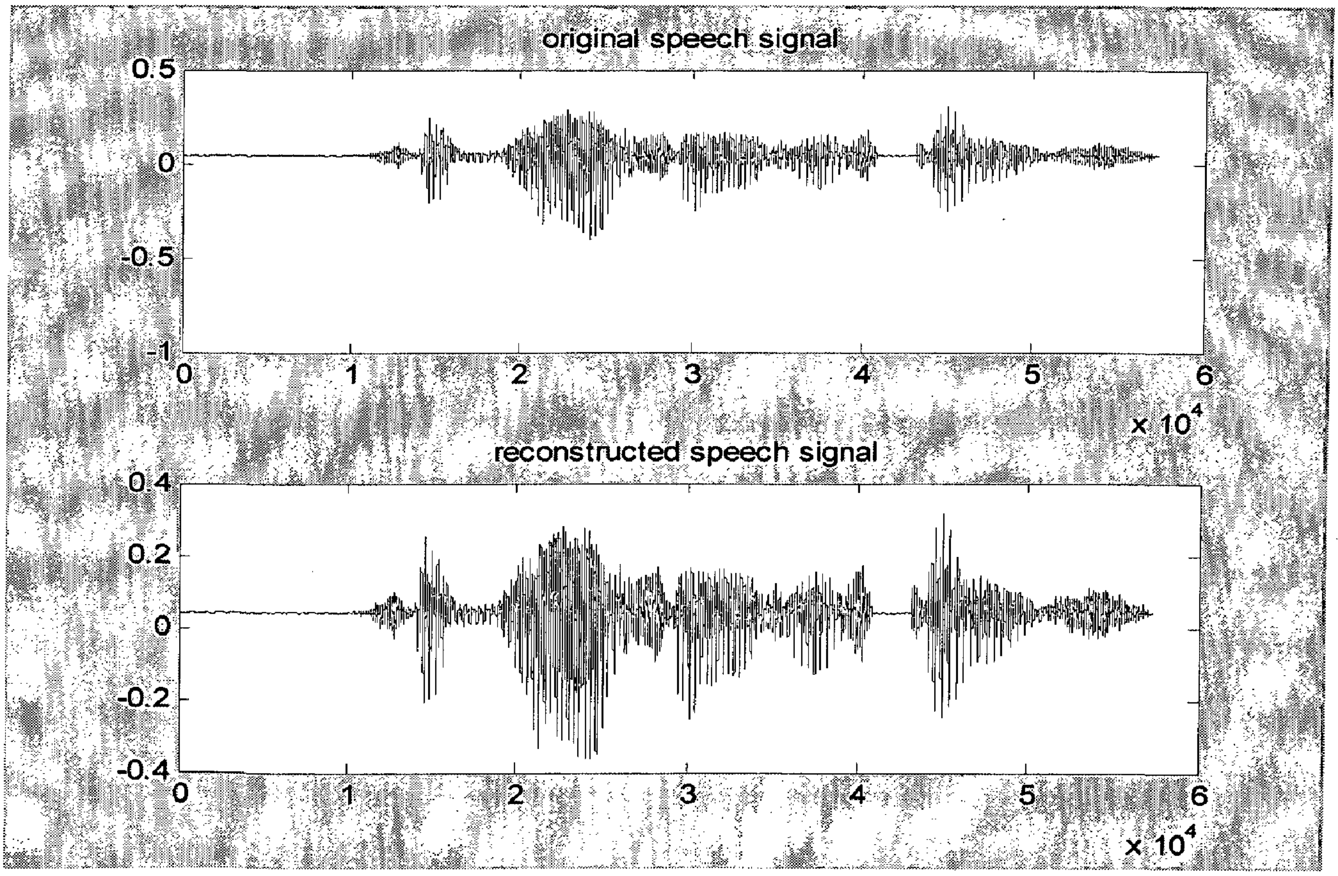


**Figure 6.6(c) Original and reconstructed speech signals ("signals and systems") of person 3 using WT technique**

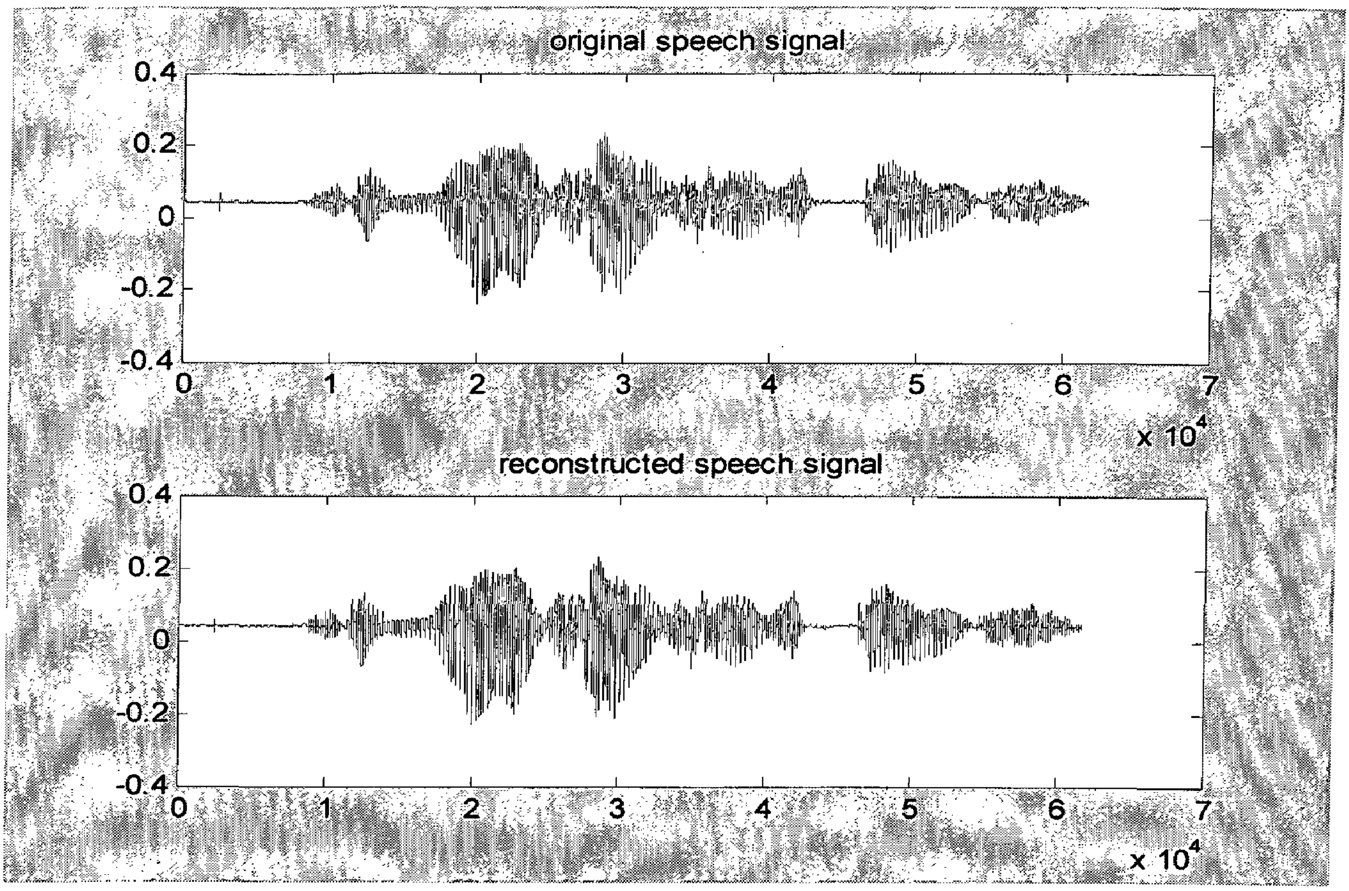


**Figure 6.6(d) Original and reconstructed speech signals ("signals and systems") of person 4 using WT technique**

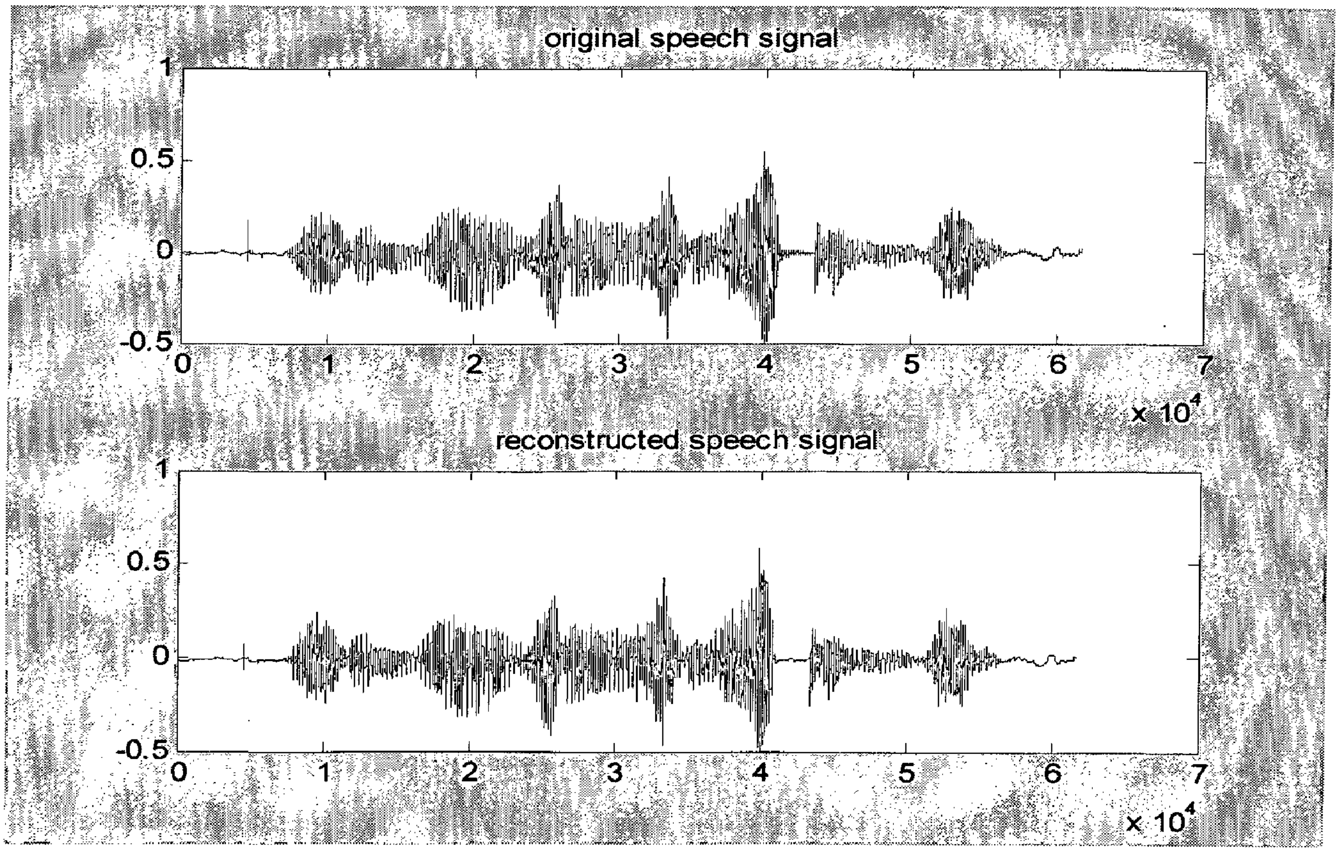


**Figure 6.6(e) Original and reconstructed speech signals ("signals and systems") of person 5 using WT technique**

Comparative tables of different parameters in the case of different compression techniques are shown in Table 6.15.

**Table 6.15 Quantitative comparison of speech compression techniques**

| Technique | LPC coding | ADPCM | WT |
|---|---|---|---|
| Comp Ratio (C) | 12.7809 | 4.3169 | 9.9587 |
| SNR | 0.2062 | 35.5209 | 17.1575 |
| PSNR | 7.1166 | 39.273 | 28.349 |
| NRMSE | 1.2968 | 0.0485 | 0.21423 |

In the Table 6.15 values of the parameters that are shown are the average. It is evident that wavelet Transform technique SNR, PSNR, and NRMSE has values in middle of LPC and ADPCM techniques.

# Chapter-7
# CONCLUSIONS AND SCOPE FOR FUTURE WORK

## 7.1 Conclusions

In this dissertation work three types of compression techniques namely LPC coding, ADPCM and Wavelet Transform has been implemented. This work mainly concentrated on compression of recorded speech signals with out significant loss of quality. Comparative evaluation was performed with respect to compression ratio, SNR, PSNR, and NRMSE. These three algorithms are tested on different speech signals spoken by different persons.

In this implementation LPC coding technique gives the average compression ratio of 12.7809 and with very less variation in compression ratio but more waveform distortion is seen. The SNR and PSNR are less with respect to other two techniques while NRMSE is more. This indicates poor quality of output speech for LPC coding.

The ADPCM technique produces average compression ratio of 4.317 and less variation of compression ratio. Waveform distortion is low in the case of ADPCM technique. The SNR and PSNR are more with respect to other two techniques and NRMSE is less. This indicates better quality output speech from this technique.

In wavelet transform based speech compression different wavelets like 'Db10', 'Haar', 'Db4', 'Db6', and 'Db8'are used for the analysis purpose. In the analysis it had shown that 'Db10' gives more compression ratio than that of all other wavelets. 'Haar' wavelet produces least compression ratio. In all other cases compression ratio is nearly equal for particular level of decomposition. Quality is more in the case of 'Db10' wavelet and less in the case of 'Haar' wavelet. The compression ratio can be changed in the case of wavelet transform based speech compression by changing the level of decomposition. At decomposition level 5 this technique is giving compression ratio of about 21 and at level 7 it is giving 37. But there is drastic change in the quality after level 3. Thus, for the comparative evaluation purpose 'db10' wavelet has been used at decomposition level 3 considering quality as an important factor.

A significant advantage of using wavelets for speech compression is that the trade off between compression ratio and quality can be achieved while for other two techniques LPC coding and ADPCM compression ratio is nearly fixed with fixed quality. In the case of wavelet transform based speech compression quality is also nearly equal with respect to ADPCM and better with respect to LPC coding. One disadvantage in the wavelet transform based speech compression is that, with respect to other two techniques this is because of much more computation involved in the wavelet transform.

## 7.2 Scope For Future Work

In this dissertation work speech compression techniques like LPC coding, ADPCM technique, and Wavelet transform has been implemented and compared with respect to compression ratio. The work can be further extended to any one of the directions:

1. For better compression ratios, further data compression ratio is possible by exploiting the redundancy in the encoded transform coefficients.
2. Implementation of the algorithms can be done by using the higher level languages like C, C++ for faster processing.
3. Wavelet transform based implementation can be done on DSP kit for real time application for speech transmission purpose.
4. In this work objective quality measuring methods like SNR are used for the comparative evaluation a subjective quality measuring methods like mean opinion score, diagnostic acceptability measure and diagnostic rhyme test can also be tried for comparison purpose.

[1] L. R. Rabiner and R. W. Schafer, "Digital Processing of Speech Signal". http://www.data-compression.com/speech.shtml

[2] Ming Yang, "Low bit rate speech coding", Potentials, IEEE, Volume 23, Issue 4, pp.32-36, Oct-Nov 2004.

[3] R. V. Cox and P. Koorn, "Low bit rate speech coders for multimedia communication", Communications Magazine, IEEE, Volume 34, Issue 12, pp. 34-41, Dec 1996.

[4] J. Hanzo, F. C. A. Somerville and J. P. Woodard, "Voice Compression and Communications", A John Wiley and Sons, INC., Publication, IEEE Press, 2001.

[5] I. Boyd, "Speech coding for telecommunications", Electronics and Communications Engineering Journal, Volume 4, Issue 5, pp.273-283, Oct 1992.

[6] A. M. Kondoz, "Digital Speech", A John Wiley and Sons, INC., Publication, 2001.

[7] J. D. Gibson, "Speech coding methods, standards, and applications", Circuits and Systems Magazine, IEEE, Volume 5, Issue 4, pp.30-49, 2005.

[8] A. G. Gandhi and S. S. Dhekane, "Speech coding at very low bit rates for mobile communication", Communications, APCC2003, the 9th Asia-Pacific Conference, Volume 1, pp.358-362, Sept 2003.

[9] http://users.ece.gatech.edu/~njayant/mmc8/sld 004.htm

[10] D. Milkovic and E. Zentner, "Speech coding methods in mobile radio communication systems", Applied Electromagnetics and Communications, ICECom 2003, 17th International Conference, pp.103-108, Oct 2003.

[11] M. Z. Markovic, "Speech compression-recent advances and standardization", Telecommunications in Modern Satellite, Cable and Broadcasting Service, TELSIKS2001 5th International Conference, Volume 1, pp. 235-244, Sept 2001.

[12] S. W. Park, M. Gomez, and R. Khastri, "Speech compression using line spectrum pair frequencies and wavelet transform", Intelligent Multimedia, Video and Speech Processing 2001. Proceedings of 2001 International Symposium, pp. 437-440, May 2001.

[13] J. Makhoul, "Linear prediction: A tutorial review", Proceedings of the IEEE, Volume 63, Issue 4, pp.561-580, April 1975.

[14] A. Graps, "An Introduction to Wavelets", Computational Science and Engineering, IEEE, Volume 2, Issue 2, pp. 50-61, summer 1995.

[15] M. Misiti, Y. Misiti, G. Oppenheim and J. Poggi, "Matlab Wavelet Tool Box", The Math Works Inc., 2000.

[16] Y. X. Zhong, "Advances in Coding and Compression", Communications Magazine, IEEE, Volume 31, Issue 7 pp.70-72, July 1993.

[17] A. Gersho, "Advances in Speech and Audio Compression", Proceedings of the IEEE, Volume 83, Issue 6, pp.900-918, June 1994.

[18] A. M. M. A. Najih, A. R. Ramli, A. Ibrahim and A. R. Syed, "Comparing speech compression using wavelets with the other speech compression schemes", Research and Development, SCORED2003 , Proceedings, Student Conference, pp.55-58, Aug 2003.

[19] M. Budagavi and J. D. Gibson, "Speech coding in mobile radio communications", Proceedings of the IEEE, Volume 86, Issue 7, pp.1402-1412, July 1988.

[20] J. I. Agbiya, "Discrete Wavelet Transform Techniques in Speech Processing", IEEE Tencon Digital Signal Processing Applications Proceedings, IEEE; pp. 514-519, Nov 1996.

[21] A. M. M. A. Najih, A. R. bin Ramli, V. Prakash and A. R. Syed, "Speech compression using discrete wavelet transform" Telecommunication Technology, NCTT2003 Proceedings, 4th national conference, pp.1-4, Jan 2003.

[22] W. Kinsner and A. Langi, "Speech and Image Signal Compression with Wavelets", WESCANEX93 Communications, Computers and Power in the Modern Environment, Conference Proceedings, IEEE, pp.368-375, May 1993.

[23] A. Chen, N. Shehad, A. Virani and E. welsh, "Discrete Wavelet Transform for Audio Compression". http://is.rice.edu/~welsh/elec431/wavelets.html

[24] M. Schindler, "Data Compression Consulting", Practical Huffman coding, http://www.compressconsult.com/ Huffman/

[25] E. B. Fgee, W. J. Phillips and W. Robertson, "Comparing Audio Compression Using Wavelets with Other Audio Compression Schemes" Canadian Conference on Electrical and Computer Engineering, Proceeding of the IEEE, Volume 2, pp. 698-701, May 1999.