Implementation of a Variable Rate Vocoder and its Performance Analysis

by

Sharath Manjunath

Thesis submitted to the faculty of the
Virginia Polytechnic Institute and State University
in partial fulfillment of the requirements for the degree of

MASTER OF SCIENCE

in

Electrical Engineering

APPROVED:

Brian D. Woerner

Dr. Brian D. Woerner, Chairman

Jeffrey H. Reed

Dr. Jeffrey H. Reed

T. Pratt

Dr. Timothy Pratt

May, 1994

Blacksburg, Virginia

Implementation of a Variable Rate Vocoder and its Performance Analysis

by

Sharath Manjunath

Committee Chairman: Dr. Brian D. Woerner

Department of Electrical Engineering

(ABSTRACT)

The use of Code Division Multiple Access (CDMA) technology for cellular radio and personal communications has ushered in a new era wherein the benefits of CDMA can be directly applied to voice transmission. In particular, the resistance to multipath and graceful degradation of performance (voice quality) in the presence of multiple access interference are the significant benefits of CDMA.

Spectral efficiency is a key issue in mobile communications. The use of vocoders reduces the average bit transmission rate for each user, thereby reducing the bandwidth necessary to transmit the speech of each user. Qualcomm-Code Excited Linear Prediction (QCELP) algorithm provides a way to implement a variable rate vocoder which could reduce the average bit rate by a factor of two.

This thesis describes a software implementation of the QCELP algorithm and the integration of the vocoder with the CDMA transmission and reception segments. A subjective performance analysis of the vocoder, using a Mean Opinion Score (MOS) test on the speech quality under various channel conditions including 1-Ray and 2-Ray Rayleigh fading environments is performed. Objective quality measures such as Signal-to-Noise Ratio (SNR), Segmental SNR (SSNR) and the Coherence Function (CF) are also investigated for comparison with the subjective analysis results.

ACKNOWLEDGEMENTS

I would like to thank Dr. Brian D. Woerner, for being my advisor and for his help, encouragement and guidance in completion of this thesis.

I would like to thank my committee members Dr. Jeffrey H. Reed and Dr. Timothy Pratt for their useful contributions and comments.

I thank Nena for all the useful discussions we had which helped me gain insight into many things which were formerly nebulous. I also thank her immensely for her constant support, encouragement and advice and assistance in timely completion of this thesis.

I also thank Rajiv for his guidance and help throughout my studies here.

I would like to thank all the listeners for their patience in assisting me in obtaining the Mean Opinion Score test results.

Lastly, I wish to thank my parents for their encouragement throughout these years of my studies here. Were it not for their support and patience I would not have had the opportunity of studying at Virginia Tech and meeting wonderful people.

TABLE OF CONTENTS

1 Introduction **1**

1.1 Speech Processing 2

1.2 Objective of Research 3

1.3 Summary 4

2 Speech Analysis and Synthesis **5**

2.1 Introduction 5

2.1.1 Speech signal 5

2.2 Waveform Coders 6

2.2.1 Time Domain Vocoders 6

2.2.2 Frequency Domain Vocoders 8

2.3 Vocoders 9

2.3.1 Cepstral Vocoder 10

2.3.2 Formant Vocoder 10

2.3.3 Linear Predictive Coding 11

2.3.4 Residual-Excited Linear Prediction 13

2.3.5 Code-Excited Linear Prediction 13

2.4 Speech Quality Assessment 17

2.5 Summary 18

3 The QCELP Algorithm **19**

3.1 Introduction 19

3.2 Encoder Model 20

3.2.1 DC Removal 21

3.2.2 Windowing 21

3.2.3 Parameters and Bit Allocations 21

3.2.4 Computing the LPC coefficients 23

3.2.5 Conversion of LPCs to LSPs 25
3.2.6 Data Rate Selection 26
3.2.7 Quantization of LSP frequencies 28
3.2.8 Interpolation of the LSPs 29
3.2.9 Pitch Parameters Determination 30
3.2.10 Codebook Parameter Determination 32
3.2.11 Packing 35
3.2.12 Decoding at the encoder 35
3.3 Decoder Model 36
3.4 Forward and Reverse Channel Models in the IS-95 standard 37

4 Simulation of QCELP **42**
4.1 Introduction 42
4.2 Simulation Procedure 42
4.2.1 QCELP Encoding and Decoding Operations 46

5 Quality Assessment of QCELP **51**
5.1 Introduction 51
5.2 Subjective Measures 51
5.3 Objective Measures 64
5.4 Summary 83

6 Conclusions **85**
6.1 Summary of the Research 85
6.2 Future Work 86

Bibliography **88**

A Execution Instructions **92**

LIST OF FIGURES

2.1 ADPCM system 7
2.2 Quadrature Mirror Filter Bank 8
2.3 Speech Model 9
2.4 RELP encoder 14
2.5 CELP Encoder 14
2.6 CELP Synthesizer 15
2.7 VSELP Decoder 16

3.1 The Encoder Model 20
3.2 Bit Allocation for a Rate 1 Frame 23
3.3 Bit Allocation for a Rate 1/2 Frame 23
3.4 Bit Allocation for a Rate 1/4 Frame 23
3.5 Bit Allocation for a Rate 1/8 Frame 24
3.6 The quantization of the LSP differences 28
3.7 The Decoder Model 36
3.8 The Reverse CDMA Channel Structure 38
3.9 The Forward CDMA Channel Structure 39

4.1 The Simulation Process 43
4.2 Input Female Speech to the Vocoder 45
4.3 Input Male Speech to the Vocoder 45
4.4 Output Female Speech - no channel 49
4.5 Error in Output Speech - Rural Environment, 1 User, 10dB SNR 49
4.6 Error in Output Speech - Rural Environment, 20 Users, 10dB SNR 50

5.1 MOS - Female Speech, Forward Channel, Rural Environment, and 10dB channel SNR 53

5.2 MOS - Female Speech, Reverse Channel, Rural Environment, and 10dB channel SNR 53
5.3 MOS - Female Speech, Forward Channel, Urban Environment, and 10dB channel SNR 54
5.4 MOS - Female Speech, Reverse Channel, Urban Environment, and 10dB channel SNR 54
5.5 MOS - Female Speech, Forward Channel, Urban Environment, and 5dB channel SNR 55
5.6 MOS - Female Speech, Reverse Channel, Urban Environment, and 5dB channel SNR 55
5.7 MOS comparison - Female Speech, Rural Environment, and 10dB channel SNR 56
5.8 MOS comparison - Female Speech, Urban Environment, and 10dB channel SNR 57
5.9 MOS - Female Speech, Forward Channel, 1 Ray Rayleigh Fading 57
5.10 MOS - Female Speech, Reverse Channel, 1 Ray Rayleigh Fading 58
5.11 MOS - Female Speech, Forward Channel, 2 Ray Rayleigh Fading 58
5.12 MOS - Female Speech, Reverse Channel, 2 Ray Rayleigh Fading 59
5.13 MOS - Male Speech, Forward Channel, Rural Environment, and 10dB channel SNR 60
5.14 MOS - Male Speech, Reverse Channel, Rural Environment, and 10dB channel SNR 60
5.15 MOS - Male Speech, Forward Channel, Urban Environment, and 10dB channel SNR 61
5.16 MOS - Male Speech, Reverse Channel, Urban Environment, and 10dB channel SNR 61
5.17 MOS - Male Speech, Forward Channel, Urban Environment, and 5dB channel SNR 62
5.18 MOS - Male Speech, Reverse Channel, Urban Environment, and 5dB channel SNR 62
5.19 MOS comparison - Male Speech, Rural Environment, and 10dB channel SNR 63

5.20 MOS comparison - Male Speech, Urban Environment, and 10dB channel SNR 63
5.21 MOS - Male Speech, Forward Channel, 1 Ray Rayleigh Fading 64
5.22 MOS - Male Speech, Reverse Channel, 1 Ray Rayleigh Fading 65
5.23 MOS - Male Speech, Forward Channel, 2 Ray Rayleigh Fading 65
5.24 MOS - Male Speech, Reverse Channel, 2 Ray Rayleigh Fading 66
5.25 MOS results for 5dB channel SNR, Reverse Channel 67
5.26 Classic SNR values for 5dB channel SNR, Reverse Channel 67
5.27 SSNR - Female Speech, Forward Channel, Rural Environment, and 10dB channel SNR 68
5.28 SSNR - Female Speech, Reverse Channel, Rural Environment, and 10dB channel SNR 69
5.29 SSNR - Female Speech, Forward Channel, Urban Environment, and 10dB channel SNR 69
5.30 SSNR - Female Speech, Reverse Channel, Urban Environment, and 10dB channel SNR 70
5.31 SSNR - Female Speech, Forward Channel, Urban Environment, and 5dB channel SNR 70
5.32 SSNR - Female Speech, Reverse Channel, Urban Environment, and 5dB channel SNR 71
5.33 SSNR comparison - Female Speech, Rural Environment, and 10dB channel SNR 71
5.34 SSNR comparison - Female Speech, Urban Environment, and 10dB channel SNR 72
5.35 SSNR - Female Speech, Forward Channel, 1 Ray Rayleigh Fading 73
5.36 SSNR - Female Speech, Reverse Channel, 1 Ray Rayleigh Fading 73
5.37 SSNR - Female Speech, Forward Channel, 2 Ray Rayleigh Fading 74
5.38 SSNR - Female Speech, Reverse Channel, 2 Ray Rayleigh Fading 74
5.39 SSNR - Male Speech, Forward Channel, Rural Environment, and 10dB channel SNR 75
5.40 SSNR - Male Speech, Reverse Channel, Rural Environment, and 10dB channel SNR 75

5.41 SSNR - Male Speech, Forward Channel, Urban Environment, and 10dB channel SNR 76
5.42 SSNR - Male Speech, Reverse Channel, Urban Environment, and 10dB channel SNR 76
5.43 SSNR - Male Speech, Forward Channel, Urban Environment, and 5dB channel SNR 77
5.44 SSNR - Male Speech, Reverse Channel, Urban Environment, and 5dB channel SNR 77
5.45 SSNR comparison - Male Speech, Rural Environment, and 10dB channel SNR 78
5.46 SSNR comparison - Male Speech, Urban Environment, and 10dB channel SNR 79
5.47 SSNR - Male Speech, Forward Channel, 1 Ray Rayleigh Fading 79
5.48 SSNR - Male Speech, Reverse Channel, 1 Ray Rayleigh Fading 80
5.49 SSNR - Male Speech, Forward Channel, 2 Ray Rayleigh Fading 80
5.50 SSNR - Male Speech, Reverse Channel, 2 Ray Rayleigh Fading 81
5.51 MOS for Female Output Speech in an Urban Environment, Forward Channel 83
5.52 CF for Female Output Speech in an Urban Environment, Forward Channel 84

LIST OF TABLES

2.1 The Five-Point Scale for MOS Testing 18

3.1 Parameters Computed by the Encoder 22
3.2 Bit Allocation for Each Parameter 24
3.3 LSP Quantization Levels 29
3.4 Codebook Gain Prediction Filter 33

5.1 The Five-Point Scale for MOS Testing 52
5.2 CF for Female Speech - Forward Channel 81
5.3 CF for Female Speech - Reverse Channel 82
5.4 CF for Male Speech - Forward Channel 82
5.5 CF for Male Speech - Reverse Channel 82

Chapter 1

Introduction

Since the dawn of time, speech has been the most popular, simple and natural mode of human communication. Because it is immediate, interactive, and is received with as little conscious effort as possible, speech fills a unique role in human interaction. The preference for voice is due to it's naturalness, freeing hands and eyes[1] for other tasks [2].

Before the invention of telephone, vocal communication was possible only among people in immediate proximity to each other. This was the primary limitation of speech. The invention of the telephone changed that.

Initially, the telephone signals were *analog* in nature. An analog message signal could be represented by a waveform which is continuous in both amplitude and time. With increased telephone traffic, the switching of analog signals became cumbersome. So telephone systems switched to the more flexible *digital* format. However, since the speech signal is inherently analog, it is converted to a digital format by *sampling* the signal at discrete instants of time and then *quantizing* the samples to make them discrete in amplitude also.

The sampling theorem [3] indicates that an analog signal may be accurately represnted by samples taken at twice the signal bandwidth. Since human speech is usually considered bandlimited to 4 kHz, the required sampling rate would be at least 8000 samples/sec. Different techniques for quantization [4] are discussed in Chapter 2, but all have the property of improving the quality of speech at the expense of greater bandwidth because the representation is more accurate when more bits per sample are used. In standard telephone systems, speech is typically represented by 8000 samples/sec and 8 bits per sample pulse code modulation (PCM) with μ-law companding. This would result in a data rate of 64

[1] Sir Richard Paget [1] is quoted to have said, "What drove man to the invention of speech was, I imagine, not so much the need of expressing his thoughts as the difficulty of 'talking with his hands full. ' It was the continual use of man's hands for craftsmanship, the chase, and the beginnings of art and agriculture, that drove him to find other methods of expressing his ideas– namely, by a specialized pantomime of the tongue and lips. "

kbits/sec. Twenty four voice channels may be time-division multiplexed together to form a standard T1 channel [5].

Wireless systems represent another evolutionary step in speech communications. While the telephone relieved voice communications of the constraints of distance, wireless systems aim to relieve the constraints of a fixed location.

Early cellular systems utilized the analog Advanced Mobile Phone Systems (AMPS) standard [6]. As system capacity became critical, a migration to digital systems commenced. Two primary standards are now available for use in North America, the IS-54 Time-Division Multiple Access (TDMA) standard [7] and the IS-95 CDMA standard [8]. Both require very efficient voice signal encoding at rates less than 10 kbps. This thesis focusses on the IS-95 standard.

1.1 Speech Processing

The first major development in speech processing was PCM [9]. This was the first attempt to represent the speech waveform in a digital format. Then the Vocoder was demonstrated by Dudley in 1939 [10]. This was basically a speech synthesizer and heralded the beginning of parametric representation of speech. The key difference between these is that waveform coders attempt to describe the speech signal directly while the vocoders characterize certain parameters of the speech signal. These are described in more detail in Chapter 2.

Digital speech processing aims to reduce the bandwidth occupied by the transmitted coded speech. Efficient use of transmission media and the restriction imposed on the radio channel capacity due to the effects of propagation require the channel bit rates to be kept as low as possible. Different methods for processing/coding speech will be discussed in the next chapter.

The mobile channel which interacts with the information in case of cellular telephony, is probably the worst channel. Various kinds of fading models such as slow fading, fast fading and frequency-selective fading [11], and various interference models like adjacent cell interference, cochannel interference, and interference from other users (in case of a CDMA system), act on the signal simultaneously, resulting in drastic consequences. It is therefore necessary to explore the interaction between the channel effects and speech coding.

Also spectral efficiency necessitates use of a lower bit rate to utilize the full channel capacity. Hence advanced speech processing techniques are necessary in place of conventional PCM. Sophisticated digital speech processing techniques remove the redundancy inherently present in speech, minimizing the transmitted bit rate while keeping the quality of speech acceptable. The fewer the number of bits transmitted, the more prone is the speech quality to degradation due to the channel errors. Efficient speech processing systems try to keep the degradation as low as possible.

Most of the speech processing techniques used in the current standards for mobile radio achieve reasonably low bit rates. Complexity, once an important issue, is now less of a deterrent, owing mainly to advances in circuit integration technologies. Hence, the primary trade-off that exists is between the quality of speech and the bit rate achievable. Speech coding techniques for wireless systems are largely based on Linear Predictive Coding (LPC), achieving acceptable speech quality at data rates around 10 kbps or less.

1.2 Objective of Research

This research will enable the completion of the simulation of the CDMA system in its entirety by providing the speech processing input module. In this way, the relationship between the channel, transmitter implementation, and voice coding in a complex CDMA system is explored. This research focuses on the implementation of Qualcomm's QCELP algorithm for a variable rate vocoder and subsequently its performance analysis for the CDMA environment. The variable rate vocoder achieves an average bit rate which is a fraction of the full rate (9600 bps). This considerably reduces the bandwidth required by each user in the CDMA system and uses the radio channel capacity efficiently. The vocoder is integrated with the IS-95 [8] transmission and reception segments in order to assess the speech quality in the CDMA environment for different channel conditions. This is explained in further detail in Chapter 4. The vocoder is implemented in software. Different assessment measures, both subjective and objective, are discussed and used to evaluate the performance of the vocoder. The simulation technique is employed to investigate the performance of the QCELP vocoder under a variety of channel conditions.

1.3 Summary

The remainder of this thesis is organized as follows. Chapter 2 discusses the different techniques available for speech processing and their relative merits and demerits. Chapter 3 describes the QCELP algorithm for implementing the variable rate vocoder. Chapter 4 focuses on the specifics of simulation. Chapter 5 describes the performance evaluation results obtained by different quality assessment techniques. Chapter 6 presents the conclusions formed about the performance of the vocoder under various situations and provides recommendations for possible future work.

Chapter 2

Speech Analysis and Synthesis

2.1 Introduction

The conversion of analog speech into digital format suitable for transmission through a communication channel after modulation is called speech coding [4]. This generally is assumed to imply a reduction in the average number of bits needed to represent a sample. Compression helps to reduce the bandwidth occupied by the information signal and hence increase the bandwidth efficiency. The conversion, however, should not introduce significant distortion into the speech when it is reproduced back from the digital format.

The different techniques for achieving the purpose outlined above are broadly classified into *waveform coding* and *voice coding*. The term *vocoders* refers to the latter class of techniques.

Waveform coding involves encoding speech directly using the time waveform or its characteristics or the spectral properties of the same. For instance, PCM or DPCM represent the speech waveform directly in terms of bit patterns corresponding to the sample amplitudes or the difference between successive samples. Voice coding, however, represents the speech by some characteristic parameters describing the speech, which in turn are encoded into appropriate bit patterns, and then estimates frames of speech from these parameters. Linear Predictive Coding (LPC) is the predominant technique to achieve this and will be discussed in further detail.

2.1.1 Speech signal

The analog speech signal is assumed to be bandlimited to 4kHz. It is then sampled at a rate of 8000 samples per second in accordance with Nyquist's Theorem (to avoid aliasing errors) [12]. In waveform coding techniques usually each of these samples is quantized and encoded separately. However, many of these samples could be quantized as a single entity

and this sequence of samples is termed as a *vector*. The index of the vector from a standard set of vectors is actually transmitted after quantization. This is called vector quantization, in contrast to scalar quantization in the former case [4]. The set of vectors is called a codebook. The primary advantage of vector quantization is that it allows exploitation of the correlation between samples. Either a waveform coder or a vocoder could use this form of coding. In the case of vocoders, the parameters could also be vector quantized. Alternatively, the speech samples could be presented through direct vector quantization. However, given today's technology, the complexity of direct vector quantization of speech is prohibitive.

2.2 Waveform Coders

As mentioned above, waveform coding could be achieved by using either the time domain characteristics or the frequency domain characteristics. Pulse Code Modulation (PCM), Differential PCM (DPCM), and Delta Modulation (DM) are the time domain waveform coding methods [9][13]. Subband Coding (SBC) and Filter Bank Coding (FBC) are frequency domain methods.

2.2.1 Time Domain Vocoders

In PCM, each sample is converted (quantized) to a corresponding bit pattern. The number of bits B, used to represent a sample is dependent on the step size used and the range of values the input speech could take. If the spacing between the 2^B levels is same, then the quantizer is called a uniform quantizer. However, usually the smaller signal amplitudes occur more frequently than the larger ones. Hence, a nonuniform quantizer would be more effective in reducing the distortion for such a signal. Lloyd [14] has shown that the optimum distribution of quantizer levels can be found [15] through iterative calculation of quantization regions and levels. The μ- law and A-law companding are popular methods for implementing nonuniform quantization. These allow implementation of a nonuniform quantizer by inserting a nonlinear device in series with a uniform Analog-to-Digital Converter (ADC) [9].

In DPCM, the inherent correlation existing between speech samples is exploited. Instead

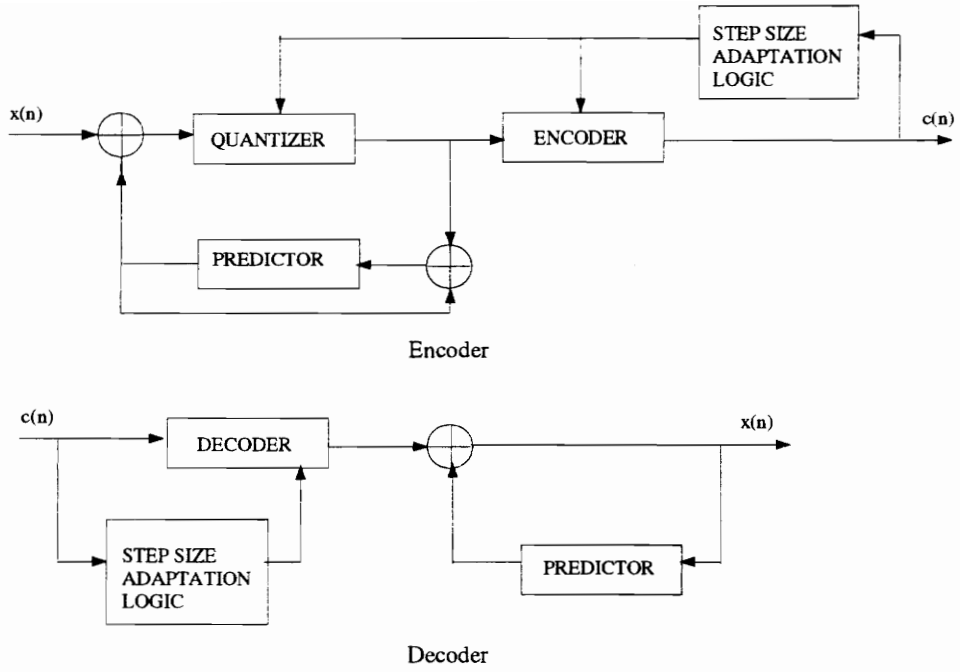

Figure 2.1: ADPCM system

of coding the samples themselves, the difference between successive samples is quantized. Since there is now a reduced range of values for the input to the quantizer, the number of bits needed to represent the coded speech is less than in conventional PCM [9].

As opposed to PCM and DPCM, Adaptive PCM (APCM) and Adaptive DPCM (ADPCM) [13] adapt to the time-variant statistics of the speech signal by adjusting the step size constantly, based either on the input or the output. Figure 2.1 shows the basic block diagram of an ADPCM coder. The predictor in an ADPCM system is a filter which decides its output based on the previous samples. ADPCM is employed as part of the Digital European Cordless Telephone (DECT) standard.

Delta Modulation (DM) [9] may be thought of as a simplified version of DPCM. A 1-bit quantizer is used instead of a larger number of bits per sample. DM systems explicitly make the choice of oversampling signals, while reducing the complexity of the quantizer. If the present sample is larger than the predicted sample the error is positive and a binary one is transmitted, else a binary zero is transmitted. As in PCM and DPCM, there is an adaptive version of DM, which greatly reduces the *slope-overload distortion* and *granular noise*. The

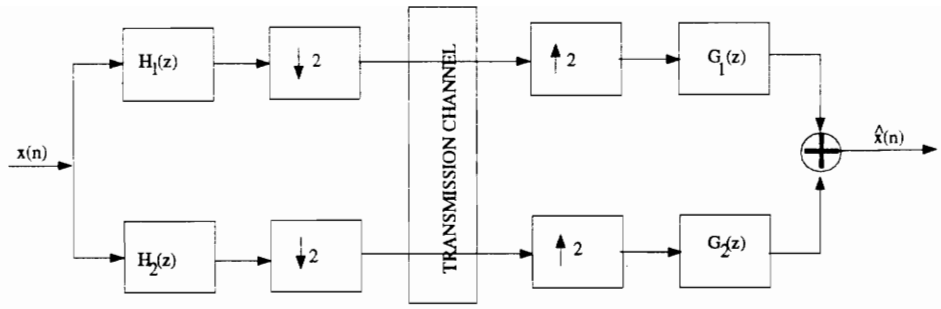

Figure 2.2: Quadrature Mirror Filter Bank

slope-overload distortion is due to the step size being too small to follow portions of the waveform that have a steep slope. The granular noise results from using a step size that is too large for portions of the waveform which have a small slope. Here again, as in ADPCM or APCM, the principle is to adapt the step size to the source signal.

2.2.2 Frequency Domain Vocoders

In Subband Coding (SBC), the signal is filtered into four to eight subbands and the waveform in each band is encoded individually. The encoding used generally is APCM, although any kind of encoding can be used. More bits are allocated for the lower frequency bands as they contain most of the energy and are the most perceptually significant. Fewer bits are used for the signals in higher frequency bands. Subband coding reduces the minimum sampling rate necessary for each band. Hence lower cost ADCs can be used for digitizing the signal. Quadrature Mirror Filters (QMF) can be used to prevent aliasing or at least reduce it to a great extent [16]. Figure 2.2 shows the basic structure of a QMF bank for a 2-band SBC. $H_1(z)$ and $G_1(z)$ are low pass filters; $H_2(z)$ and $G_2(z)$ are high pass filters. $\downarrow 2$ indicates downsampling by a factor of 2, whereas $\uparrow 2$ indcates upsampling by a factor of 2. The QMFs could be built entirely from simple allpass sections with a minimum number of multipliers.

In Filter Bank Coding (FBC), the frames of speech, N samples long, are divided into N bands, the frequency of each corresponding to integral multiples of $\frac{2\pi\omega_s}{N}$, where ω_s is the sampling frequency in rad/s. The signal in each band corresponds to the samples of the complex spectrum of the frame evaluated using the short-term Discrete Fourier Transform

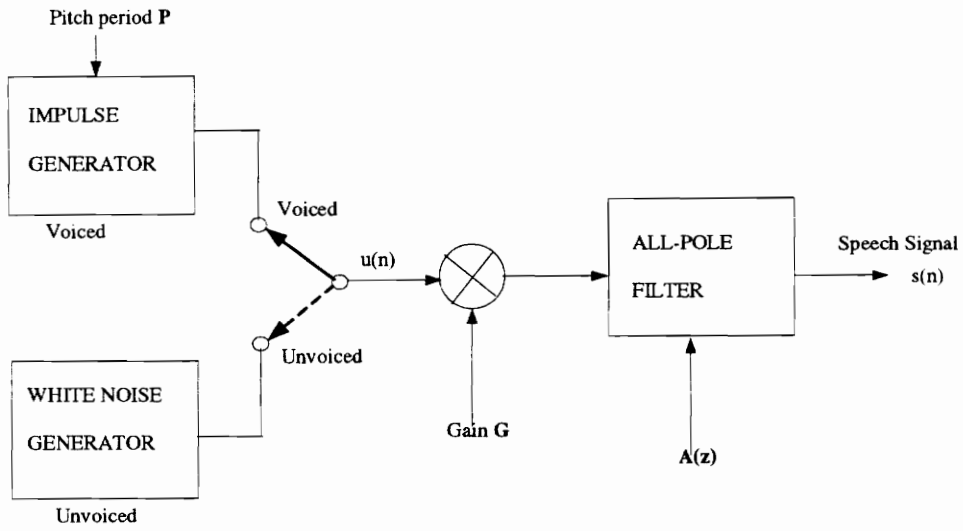

Figure 2.3: Speech Model

(DFT). More detailed discussion of waveform coding techniques can be found in [17].

In all the above cases, vector quantization could be used instead of scalar quantization, further reducing the bit rate.

2.3 Vocoders

The waveform coding techniques described in the previous section are implemented on a sample or frame basis with the speech waveform represented directly in the time and frequency domain. Vocoders, in contrast, represent the signal by an all-pole model of the vocal tract. A high level representation of the model is shown in Figure 2.3.

There are basically two types of speech sounds, 'voiced' and 'unvoiced'. Voiced sounds are produced by forcing air through the glottis and the vocal cords vibrate in an oscillatory mode. This causes quasi-periodic puffs of air to excite the vocal tract. Hence, the waveforms of voiced sound are characterized as deterministic [17]. Unvoiced sounds are caused by a turbulent noiselike random excitation of the vocal tract. The speech signal's excitation (which excites the filter of Figure 2.3) is assumed to be periodic if the signal is voiced and is assumed to be random white noise if it is unvoiced. The vocoder estimates the model parameters by analyzing the frames of speech, encodes and transmits them on a frame-by-

frame basis and reconstructs the speech signal from these parameters at the receiver. Higher bandwidth efficiency is usually possible with vocoders. Typical data rates of under 9600 bps are possible for toll-quality speech. The vocoder implementation in this thesis gives toll-quality speech at an average data rate of approximately 6000 bps. It uses a variation of the Code Excited Linear Prediction (CELP) algorithm which is described in Section 2.3.5. Several broad classes of vocoders have been proposed.

2.3.1 Cepstral Vocoder

The voiced speech as in Figure 2.3 is assumed to be generated by a periodic sequence of impulses applied to the vocal tract filter, and unvoiced speech is assumed to be generated by a white noise sequence. The excitation in both cases varies faster than the output of the filter.

The *cepstrum* is a transformation of a signal (particularly voiced speech) [18] which gives an insight into the component parts of the signal if they are convolved in the time domain. The spectrum is no longer useful in this case since it is not separable into the components of a signal unless they are linearly combined. The real cepstrum $c(n)$ is computed as

$$c(n) = \mathcal{F}^{-1}\left\{\log|\mathcal{F}\{x(n)\}|\right\} \tag{2.1}$$

where $\mathcal{F}\{.\}$ denotes the DFT and $x(n)$ is the input signal.

If the frames of speech are processed in the cepstral domain, the slowly varying vocal tract spectrum can be separated from the faster periodic spectrum due to the pitch. Hence, after separating the spectra, the characteristics of the vocal system can be estimated. This kind of vocoder is called the Cepstral Vocoder. Further information on cepstral vocoders can be found in [19].

2.3.2 Formant Vocoder

The formant vocoder estimates the first few (3-4) *formants* and their bandwidths for a speech frame. A formant is a resonance frequency of the vocal tract. The formants and the pitch period are encoded and transmitted. The formants can be estimated by linear prediction or using the cepstrum. However, it is difficult to obtain estimates of the formants when two of them are very close to each other. This problem has caused the formant vocoder

to be of limited use. An advantage of this vocoder is that large bandwidth efficiencies are possible because very few parameters are encoded per frame. Bit rates as low as 600-800 bps are therefore possible. Speech quality will be excellent if the formants are estimated accurately [20].

2.3.3 Linear Predictive Coding

Linear prediction estimates the parameters of an all-pole model of the vocal tract. The excitation type, the pitch period and the gain also have to be estimated. These parameters are then quantized, some of them after being compressed in range through a logarithmic transformation. A large number of bits must be allocated to the prediction coefficients because the filter model is very sensitive to small changes in their values.

Basic Principles

The all-pole digital filter representing the vocal tract is a time-varying filter having a steady state system transfer function of the form

$$H(z) = \frac{S(z)}{U(z)} = \frac{G}{1 - \sum_{k=1}^{M} a_k z^{-k}} \tag{2.2}$$

where G is the gain, $\{a_k\}$ are the filter coefficients and M is the order of the filter. $S(z)$ and $U(z)$ are the z-transforms of the speech signal and the excitation respectively. As discussed before, this system is excited by an impulse sequence for voiced speech and random noise for unvoiced speech.

For the model in Figure 2.3, the speech samples $s(n)$ and the excitation $u(n)$ are related by [20]

$$s(n) = \sum_{k=1}^{M} a_k s(n-k) + Gu(n) \tag{2.3}$$

A linear predictor with prediction coefficients, α_k gives an output

$$\tilde{s}(n) = \sum_{k=1}^{M} \alpha_k s(n-k) \tag{2.4}$$

The prediction error is then defined as

$$e(n) = s(n) - \tilde{s}(n) = s(n) - \sum_{k=1}^{M} \alpha_k s(n-k) \tag{2.5}$$

Hence, the prediction error is the output of a system whose transfer function is

$$A(z) = 1 - \sum_{k=1}^{M} \alpha_k z^{-k} \tag{2.6}$$

If the model accurately describes the speech signal, and $\alpha_k = a_k$, then $e(n) = Gu(n)$. Thus $A(z)$, which is the prediction error filter, will be the inverse filter for the system $H(z)$, i. e. , from Equation 2.2,

$$H(z) = \frac{G}{A(z)} \tag{2.7}$$

Thus, LP analysis consists of the determination of the set of coefficients $\{\alpha_k\}$ from the speech signal. Because of the time-varying nature of speech, these coefficients are estimated for short segments of the speech signal. A set of predictor coefficients is found such that the mean-squared prediction error is minimized over a short segment of the speech signal.

The gain G, is computed by minimizing the difference between the error signal energy and the energy in the excitation input, i. e. , G is found such that $G^2 \sum_{m=0}^{N-1} u^2(m) - \sum_{m=0}^{N-1} e^2(m)$ is minimum. Normally, for voiced speech, it is reasonable to assume that $u(n) = \delta(n)$, where $\delta(n)$ is the unit impulse function. For unvoiced speech, it is a good assumption that $u(n)$ is a zero-mean, unit variance, stationary, white noise process.

The LPC coefficients are found using one of many methods including the covariance method, autocorrelation method, Levinson-Durbin recursion for the autocorrelation method, etc. The algorithm used in this thesis implements the Levinson-Durbin recursion [17] to calculate the LPC coefficients.

If a different set of parameters for the filter model is chosen, such as the Line Spectrum Pairs (LSP) [21], explained in detail in Chapter 3, or the reflection coefficients, then the dynamic range is may be reduced, and a smaller number of bits can be used to represent the signal. The reflection coefficient is mathematically defined as the ratio of the difference in areas of consecutive tubes to their sum in a cocatenated lossless tube model. An estimate of these reflection coefficients can be obtained as a by product in the computation of the LPC coefficients. The relation between the reflection coefficients and the LP coefficients is explained in Chapter 3. Sometimes the reflection coefficients are nonlinearly transformed by either the inverse sine transform into

$$\sigma_k = \frac{2}{\pi} \sin^{-1} r_k, \quad 1 \leq k \leq M, \tag{2.8}$$

where r_k is the k^{th} reflection coefficient and M is the order of the predictor. Alternatively, the log-area ratio (LAR) transform may be used to transform the reflection coefficients as shown in Equation 2.9.

$$l_k = \tanh^{-1} r_k, \quad 1 \leq k \leq M. \tag{2.9}$$

This does not complete the LPC design however, since the excitation sequence must be generated for synthesizing the speech at the receiver. Different analysis and synthesis schemes for speech coding result in different types of excitation signals.

2.3.4 Residual-Excited Linear Prediction

Residual-Excited Linear Prediction (RELP) [22] is one of the more common LPC methods. RELP is employed in the European Group Speciale Mobile (GSM) standard for mobile communications. In this method, the residual error is transmitted to the receiver. The residual error is defined as the difference in synthesized speech and actual speech and may be calculated by subtracting the synthesized speech (based on the LPC model and excitation parameters estimated from the current frame of speech) from the original speech signal. This error is quantized and coded and transmitted to the receiver with the model parameters. The synthesis is performed by adding the error to the response of the model.

The RELP vocoder calculates the residual error by passing the speech signal through the inverse filter $A(z)$. This is bandlimited by a low pass filter, decimated, transformed to the frequency domain by the DFT and encoded. At the receiver, the received signal is transformed back to the time domain, interpolated and filtered back. The higher frequency components are then recovered by passing the interpolated received signal through a rectifier and a high pass filter, which are then added to the lower frequency components. No pitch and voicing information is needed in this case. The block diagram of the RELP encoder is shown in Figure 2.4.

2.3.5 Code-Excited Linear Prediction

An improved selection of the excitation sequence reduces the bit rate further as in the case of Code-Excited Linear Prediction (CELP) [23]. The standard CELP algorithm was developed by NSA for 4000 bits/sec encoding of voice signals. This is an analysis-by-

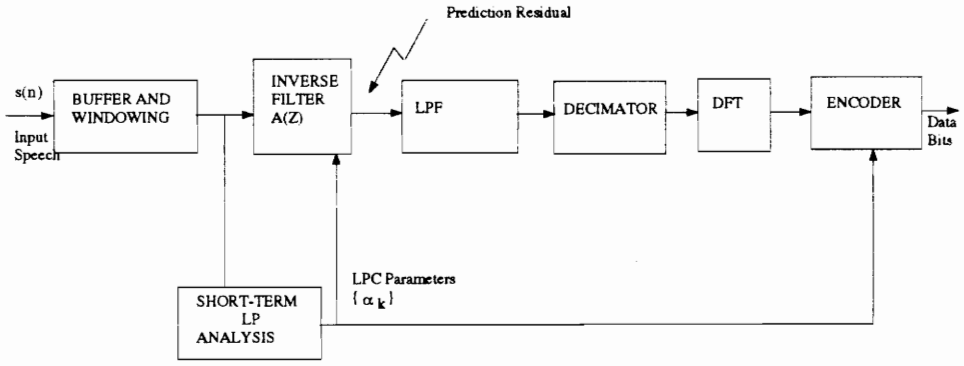

Figure 2.4: RELP encoder

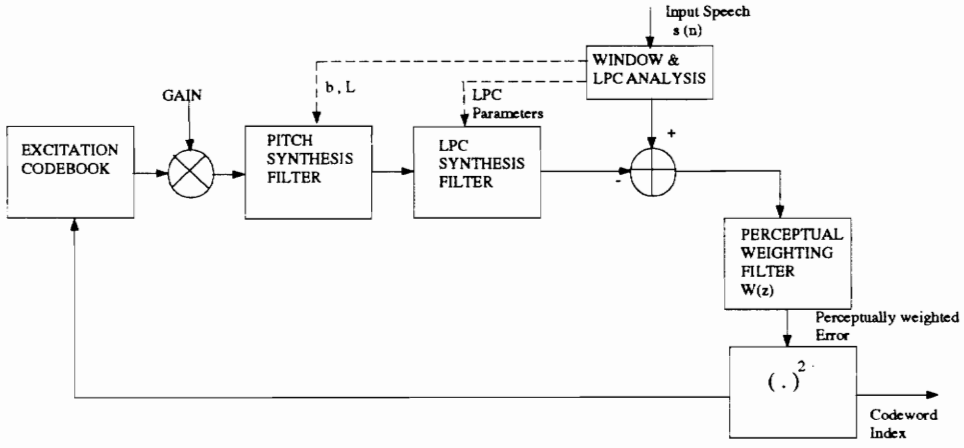

Figure 2.5: CELP Encoder

synthesis method where the excitation sequence is selected from a codebook of zero-mean Gaussian sequences [17]. The CELP encoder and synthesizer are shown in Figures 2.5 and 2.6. The CELP synthesizer has two all-pole filters in tandem, with coefficients that are updated periodically. The first is a long-delay pitch filter used to generate the periodicity in speech. This filter has the form

$$\frac{1}{1 - bz^{-L}},$$

where b and L are determined by minimizing the prediction error energy after pitch estimation, for a frame of duration 5 msec. The second filter represents the short term characteristics of the speech and is used to generate the formants of the speech signal. Usually the coefficients of this filter (10-12 in number) are determined periodically, every 15-20

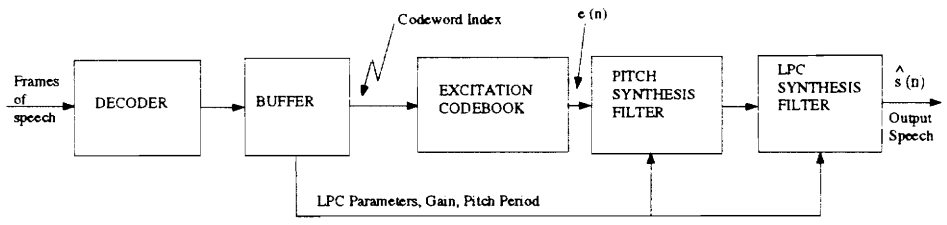

Figure 2.6: CELP Synthesizer

msec.

The CELP coder is of the analysis-by-synthesis type. The input speech is divided into segments of length 15-20 msec, which are called frames. The frames themselves would be subdivided into subframes which are speech segments of shorter duration than the frame. The pitch period and the codebook paramters are computed for every subframe in every frame. The LPC coefficients are computed only once per frame. A stored sequence from an excitation codebook is scaled and used as excitation for the filter cascade of a pitch synthesis filter and an LPC synthesis filter (of the current frame). This synthetic speech is compared with the current frame of speech and the difference (residual error) is perceptually weighted by a system whose transfer function is

$$W(z) = \frac{A(z)}{A(z/\xi)}. \tag{2.10}$$

where $A(z)$ is described by Equation 2.6 and ξ is in the range (0,1] and controls the weighting of the noise spectrum.

The sum-squared error, which is the error energy for the current subframe within a frame, is minimized over all possible excitation sequences in the codebook. The gain factor is found likewise for each codeword. Then, the index of the codeword in the codebook which minimized the error energy, is transmitted along with the gain. Both of these values are quantized, the latter after it is logarithmically compressed.

The Vector Sum Excited Linear Prediction (VSELP) vocoder [24] is a variation of the conventional CELP vocoder. The VSELP decoder is shown in Figure 2.7. The major difference between CELP and VSELP is that the latter has three excitation sources as compared to one in the former. The three excitation sources are the long-term (pitch) predictor state, and two codebooks each with some specified number of codewords, respectively. The pro-

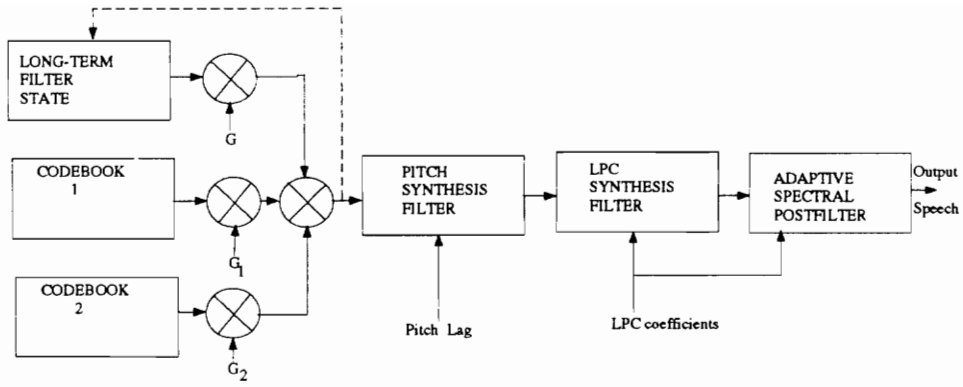

Figure 2.7: VSELP Decoder

cess of determination of the excitation sequences from these sources is same as in CELP. The LPC synthesis filter and the pitch filter are the same as those in CELP. The former's coefficients are transmitted every 20 msec and are updated every 5 msec by interpolation. The codebook parameters are also updated every 5 msec. The codewords in each of the codebooks are constructed from two sets of basis codewords (seven in number) by forming their linear combinations. Each of the basis codewords are usually selected to be like the ones in the CELP codebook. The VSELP algorithm is employed in the North American Digital Cellular (IS-54) Standard [7]. The long-term state which is the memory of the pitch filter, is also a codebook, where the codewords correspond to certain lags or pitch periods of the pitch filter. The three excitation sequences are selected sequentially and each codebook search tries to minimize the total perceptually weighted error energy. The gains G, G_1 and G_2 which minimize the error energy are found. All these parameters are vector quantized and transmitted every 5 msec. The pitch lag is sent every 5 msec, speech energy every 20 msec, and the LPC coefficients (reflection coefficients) every 20 msec. An adaptive spectral postfilter is used after the LP synthesis filter, and is of the form

$$PF(z) = \frac{B(z)}{A(z/\xi)} \tag{2.11}$$

where the coefficients of $B(z)$ are found to smooth out the speech spectrum, and ξ is similar to that in Equation 2.10, used to weight the speech spectrum. A detailed description of VSELP is provided in [24] and [25].

2.4 Speech Quality Assessment

There is always a need to assess the speech quality when compression techniques such as those described above are used, since the speech itself is not transmitted; the parameters representing the speech characteristics are transmitted instead. How well these parameters represent the speech is evaluated using speech quality assessment techniques.

With compression of speech, there is an intrinsic trade-off between the bit rate and speech quality. The larger the bit rate, the greater the information about the speech that is transmitted, and the better the speech quality.

However, there is also another problem in evaluating the speech, namely, standardization since speech evaluation must ultimately be based on individual perception. Since considerable difference exists in perceptions of different individuals, it is very necessary to have standardized procedures to evaluate speech. These procedures are collectively termed *subjective quality measures*. There is also the possibility of resorting to *objective measures*, which are sometimes capable of giving a good insight into the system performance. These tests measure the distortion between the input and output of coding systems, and include signal-to-noise ratio and LPC-based distances. The measure which most closely predicts the subjective quality is used eventually, though there is no universal consensus as to the most accurate quality measure. The most common of all subjective measures is the Mean Opinion Score (MOS) test and the most popular among all objective measures is the Signal-To-Noise Ratio (SNR). MOS testing primarily involves collecting opinions from various listeners as to the quality of speech. The opinions take the form of a numerical grade on the speech outputs. The most popularly used grading scheme is the five-point scale. This scale consists of grades (scores) ranging from 5 down to 1. The relation between the scores and the speech quality is outlined as in Table 2.1 [17].

The opinion scores obtained by each listener are then averaged to get the mean score for each speech output. Hence the name, Mean Opinion Score. These measures are used in this thesis to evaluate the performance of the speech coding system and are further discussed in Chapter 4.

Table 2.1: The Five-Point Scale for MOS Testing

Score	Speech Quality	Level of Distortion
5	Excellent	Imperceptible
4	Good	Just perceptible but not annoying
3	Fair	Perceptible and slightly annoying
2	Poor	Annoying but not objectionable
1	Unsatisfactory	Very annoying and objectionable

2.5 Summary

This chapter described some of the various ways of representing and/or compressing speech in a form suitable for transmission. Linear predictive techniques such as CELP, VSELP, and other variations of CELP are the most popular among the speech compressions methods used, though there are systems which use RELP too. These methods could give the lowest bit rates possible while providing good quality of speech. Quality measures were also briefly discussed and subjective measures were declared to be preferable for the purpose.

The next chapter describes the algorithm used in this thesis, which is a variation of the CELP method, and is termed QCELP (Qualcomm-CELP). Chapter 4 contains the simulation results and also the results of performance evaluation of the vocoder in various channel conditions.

Chapter 3
The QCELP Algorithm

3.1 Introduction

The vocoder described for use with the IS-95 standard implements a variation of the CELP algorithm. As described in Section 2.3.5, this technique uses codewords from a codebook to quantize the residual error signal using an analysis-by-synthesis method. The vocoder also produces variable data rate based on voice activity. The rates possible are 9600 bps (Rate 1), 4800 bps (Rate 1/2), 2400 bps (Rate 1/4) and 1200 bps (Rate 1/8) [26]. The average data rate is reduced from the full data rate by a factor of 1.5 to 2 for typical telephone speech due to the natural pauses in human speech and periods of silence on either end in two way conversation.

The QCELP algorithm exploits the voice activity and causes the transmitter to reduce its output power level for 10 ms when a 4800 bps frame is transmitted, for 15 ms when the data rate is 2400 bps, and for 17.5 ms when the data rate is 1200 bps. During these periods, there is very less interference to other mobile stations. Thus, the average interference level is reduced. Further, since the data rate is varying, the average information rate is a fraction of the full rate, 9600 bps. Hence, the bandwidth occupied by the signal from a user is lowered, thus causing an improvement in the capacity of the CDMA system.

The speech encoder determines the parameters which minimize the perceptual difference between the synthesized output speech and the input speech. Then these parameters are quantized and packetized for transmission. The encoder contains a part of the synthesizer used to update the memories of the filters. The decoder unpacks the data, unquantizes the parameters and reconstructs the speech from them by filtering the generated codebook vector.

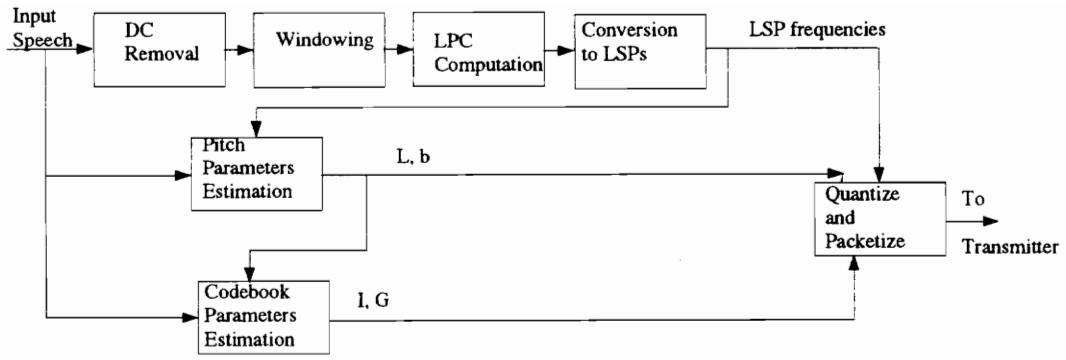

Figure 3.1: The Encoder Model

3.2 Encoder Model

The speech signal after being bandlimited to 4 kHz, is sampled at 8 kHz (Nyquist Rate). The sampled speech is split into 20 msec frames (each frame consists of 160 samples). Each sample is quantized to a uniform PCM format with 16 bits and then scaled by a factor of 4 to have the 14-bit 'integer input quantization' needed to implement this algorithm, as explained in [26].

The block diagram of the encoder is shown in Figure 3.1. The input speech frame is rendered free of DC in order that the rate decision is not adversely affected because of increase in the speech energy due to the DC. The DC is removed digitally as explained in Section 3.2.1. This speech is windowed by a Hamming window and then the LPC coefficients are found for this windowed frame of speech, from which the Line Spectral Pair (LSP) frequencies (a transformation of the LPC coefficients explained in Section 3.2.5) are calculated. The data rate is then found based on the current estimate of background noise and speech energy for the frame. Each frame is divided into subframes. The subframes which are used to compute the pitch gain and lag are called pitch subframes. The subframes used to compute the codebook gain (the gain with which the excitation sequence is to be scaled before passing it through the synthesis filters) and codebook index (the index of the codeword or vector in the set of sequences called the codebook) are called codebook subframes. There are two codebook subframes in every pitch subframe. There are 4 pitch subframes in a Rate 1 frame, 2 in a Rate 1/2 frame and 1 in a Rate 1/4 frame. There are

no pitch subframes in a Rate 1/8 frame as pitch parameters are not computed for this rate. Pitch lag and gain are found for each pitch subframe and the codebook parameters, the index and the gain are found for each codebook subframe. The pitch synthesis filter, the LPC synthesis filter, and the perceptually weighted filter present in the decoder's version at the encoder, are updated with these values. In case of Rate 1/8 (1200 bps) at which background noise is encoded, there is only a single codebook subframe, and these filters are updated using the 16-bit packet itself as the seed for the pseudo-random number generator.

3.2.1 DC Removal

The DC is removed by subtracting the low-pass filtered average of the 160 samples of the current speech frame, from the unfiltered current frame. The low-pass filtering action is achieved by adding 0.875 times the previous value of the average to 0.125 times the current value. This action produces a first order low-pass filter with the transfer function

$$\frac{0.125}{1 - 0.875z^{-1}}$$

3.2.2 Windowing

A Hamming window centered at the 140^{th} sample of the frame, is used to window the speech after removing DC from it. If $s(n)$ is this speech signal, then the windowed speech signal is given by

$$s_w(n) = s(n + 60)W_H(n), \quad 0 \leq n < 160, \tag{3.1}$$

where $W_H(n)$ is the Hamming window sequence, given by

$$W_H(n) = 0.54 - 0.46 \cos\left(\frac{2\pi n}{N-1}\right), \quad 0 \leq n < N, \tag{3.2}$$

where N is the length of the Hamming window (here $N = 160$).

3.2.3 Parameters and Bit Allocations

Table 3.1 summarizes the different parameters computed by the encoder prior to transmission. Figures 3.2 to 3.5 show the bit allocations for each data frame. In each of these figures, LPC frame means the windowed speech frame and consists of 10 LSP coefficients

Table 3.1: Parameters Computed by the Encoder

Parameter	Definition
LSP	Line Spectral Pair frequencies, 10 in number, are computed for the windowed speech frame from the LPC coefficients for that frame
PLAG	The Pitch Lag for every pitch subframe. This denotes the extent of dependence of the current sample on the previous samples
PGAIN	This is the Pitch Gain or the gain of the pitch prediction filter, and is estimated for every pitch subframe
CBINDEX	Codebook Index which indicates the code vector for excitation corresponding to the least error in the synthesized signal. This is computed once every codebook subframe, but not for Rate 1/8
CBGAIN	This is the Codebook Gain which is used to scale the excitation sequence at the synthesizer
CBSEED	This is a pseudorandom seed of 4 bits which in conjunction with the LSP bits and the CBGAIN bits gives a seed for generation of a pseudorandom sequence as the excitation vector at the synthesizer

LPC Frame	40							
Pitch Subframe	10		10		10		10	
Codebook Subframe	10	10	10	10	10	10	10	10

Figure 3.2: Bit Allocation for a Rate 1 Frame

LPC Frame	20			
Pitch Subframe	10		10	
Codebook Subframe	10	10	10	10

Figure 3.3: Bit Allocation for a Rate 1/2 Frame

for all rates. The pitch subframe consists of the pitch gain and lag, while the codebook subframe consists of the codebook index (random seed in case of Rate 1/8) and the codebook gain.

The LSPs are 10 in number for every frame. The pitch gain and lag are computed four times for a Rate 1 frame, twice for a Rate 1/2 frame, once for a Rate 1/4 frame. The codebook index (CB Index) and codebook gain (CB Gain) are determined eight times for a Rate 1 frame, four times for a Rate 1/2 frame, twice for a Rate 1/4 frame. The codebook seed (CB Seed) is generated for the Rate 1/8 frame. This seed is used in the decoder to generate a pseudorandom sequence to be used as excitation.

3.2.4 Computing the LPC coefficients

The autocorrelation function R_k, k being the shift is calculated for the current frame of speech as

$$R_k = \sum_{m=0}^{160-1-k} s_w(m)s_w(m+k), \qquad 0 \leq k \leq 10. \tag{3.3}$$

The LPC coefficients are then calculated using the Levinson-Durbin Recursion method

LPC Frame	10	
Pitch Subframe	10	
Codebook Subframe	10	10

Figure 3.4: Bit Allocation for a Rate 1/4 Frame

LPC Frame	10
Pitch Subframe	0
Codebook Subframe	6

Figure 3.5: Bit Allocation for a Rate 1/8 Frame

Table 3.2: Bit Allocation for Each Parameter

Parameter	Full Rate	Half Rate	Quarter Rate	Eighth Rate
LSP	40	20	10	10
Pitch Lag	7	7	7	-
Pitch Gain	3	3	3	-
CB Index	7	7	7	-
CB Gain	3	3	3	-
CB Seed	-	-	-	6

on the autocorrelation function. The algorithm is described as follows.

$$E^0 = R_0$$

$$\text{for} \quad i = 1, \ldots, M$$

$$k_i = \frac{1}{E^{i-1}} \left\{ R_i - \sum_{j=1}^{i-1} \alpha_j^{i-1} R_{i-j} \right\}$$

$$\alpha_i^i = k_i$$

$$\alpha_j^i = \alpha_j^{i-1} - k_i \alpha_{i-j}^{i-1}, 1, \ldots, i-1$$

$$E^i = E^{i-1} \left[1 - k_i^2 \right]$$

$$i = i + 1$$

$$\text{end}$$

where M is the order of the predictor and is equal to 10, α_j^i is the j^{th} LPC coefficient for predictor order i, E^i is the error energy associated with an order-i predictor, and k_i is the i^{th} reflection coefficient. The LPC coefficients are then scaled as follows to achieve bandwidth expansion.

$$a_i = \beta^i \alpha_i, \quad 1 \leq i \leq M, \tag{3.4}$$

where β is 0.9883.

3.2.5 Conversion of LPCs to LSPs

The Line Spectrum Pairs (LSP) are determined by using the LPC prediction filter

$$A(z) = 1 - a_1 z^{-1} - a_2 z^{-2} - \cdots - a_{10} z^{-10} \tag{3.5}$$

Now, $A(z)$ can be split into two polynomials of order 11 given by

$$\begin{aligned} P(z) &= A(z) + z^{-11} A(z^{-1}) \\ &= 1 + p_1 z^{-1} + \cdots + p_5 z^{-5} + p_5 z^{-6} + \cdots + p_1 z^{-10} + z^{-11} \qquad (3.6) \\ Q(z) &= A(z) - z^{-11} A(z^{-1}) \\ &= 1 + q_1 z^{-1} + \cdots + q_5 z^{-5} - q_5 z^{-6} - \cdots - q_1 z^{-10} - z^{-11} \qquad (3.7) \end{aligned}$$

The LSP frequencies are then the ten roots which exist between $\theta = 0$ and $\theta = \pi$ of the following equations.

$$P(\theta) = \cos(5\theta) + p'_1 \cos(4\theta) + \cdots + p'_4 \cos(\theta) + p'_5/2 \tag{3.8}$$

$$Q(\theta) = \cos(5\theta) + q'_1 \cos(4\theta) + \cdots + q'_4 \cos(\theta) + q'_5/2 \tag{3.9}$$

where

$$\begin{aligned} p'_0 &= 1 \\ p'_i &= p_i - p'_{i-1}, \quad 1 \le i \le 5 \\ q'_0 &= 1 \\ q'_i &= q_i + q'_{i-1}, \quad 1 \le i \le 5. \end{aligned}$$

The last transformation is possible since the roots of $P(z)$ and $Q(z)$ all lie around the unit circle, in complex conjugate pairs as the both polynomials have real coefficients. Only five roots each of both polynomials suffice to convey the information about all the roots, and since they lie around the unit circle, only their angle θ need be stored or transmitted. The LSP frequencies are precisely these angles (the values used for transmission are $\omega = \theta/(2\pi)$). They occur in pairs since every root of $P(z)$ has a corresponding root for $Q(z)$ placed immediately next to it on the unit circle, and these roots are interleaved. The fact that the roots of these two polynomials are interleaved on the unit circle can be of great advantage when solving for them from the above equations. An interesting thing to note here is that

these pairs of zeros correspond to pairs of poles on the unit circle in the synthesis filter, representing an undamped sine wave which has a line spectrum; and hence these frequencies are called Line Spectrum Pairs. A detailed description of the properties of LSPs is given in [21].

In this implementation of QCELP, a slight difficulty was encountered in solving for these roots. This was due to numerical precision problems. Initially, a non-linear least-squares optimization method (function **fsolve** of MATLAB) was used to solve these equations, since they are transcendental. One problem with this approach was that initial conditions were needed for every root and these did not necessarily give different solutions. Another disadvantage of this approach was that the interleaving of the roots was not exploited. Although the roots of one equation could be used as initial conditions for the other, there were no bounds on the roots of the latter and hence could possibly produce a root which is the same as a previously determined one. Hence each equation had to be solved individually. In order to alleviate the complications of a non-linear optimization technique for a transcendental set of equations, the Laguerre's method [27] was used to compute the roots of the polynomial in z^{-1} and the angles of the roots as the LSPs. However, this method too faced the problem of supplying initial conditions which could become tedious and cumbersome. In addition, complex numbers were used in the computation and due to inaccuracies in the representation, the magnitude of the roots was sometimes found to differ from one. Therefore, this second method could not be very successfully implemented. Finally, the classic method of bisection was implemented. Here, the unit circle is divided into a number of elemental arcs. Then, the function whose roots are to be found is tested to have a sign change in these arcs by bisecting the arc into two parts successively. If there is a sign change in any portion of the arc, then the remaining portion is discarded and the point of the sign change is closed in upon by again bisecting successively. This worked admirably well and also could exploit the interleaving of the roots since the bisection method makes use of the end points of the interval in which the root is supposed to lie.

3.2.6 Data Rate Selection

The data rate at which the frame is to be transmitted is decided by the frame energy and its comparison with three thresholds. The energy in the frame is estimated by R_0 which is

the autocorrelation function at zero shift. The three thresholds are calculated based on the estimate of the background noise level, B_i of the i^{th} frame. They are a quadratic function of the background noise estimate. A maximum limit is set on B_i to be equal to 5059644 as specified in [26] so that the input speech frame of sufficient energy is not encoded as background noise. The background noise is estimated for the current frame as follows.

$$B_i = \min\left(R_{0,\text{prev}}, 5059644, \max\left(1.00547B_{i-1}, B_{i-1}+1\right)\right) \tag{3.10}$$

This means that the noise estimate is increased at least by one from that in the previous frame, but at the same time making sure that it doesn't exceed the energy in the previous frame $R_{0,\text{prev}}$ (which itself is approximately an estimate of background noise) or the maximum limit.

The thresholds are calculated as follows for every frame depending on whether the noise estimate is above or below 160000.

If $B_i > 160000$

$$\begin{aligned} T_1(B_i) &= -9.043945\times10^{-8}B_i^2 + 3.535748B_i + 62071 \\ T_2(B_i) &= -1.986007\times10^{-7}B_i^2 + 4.941658B_i + 223951 \\ T_3(B_i) &= -4.838477\times10^{-7}B_i^2 + 8.63002B_i + 645864 \end{aligned} \tag{3.11}$$

else

$$\begin{aligned} T_1(B_i) &= -5.544613\times10^{-6}B_i^2 + 4.047152B_i + 362 \\ T_2(B_i) &= -1.529733\times10^{-5}B_i^2 + 8.750045B_i + 1136 \\ T_3(B_i) &= -3.95705\times10^{-5}B_i^2 + 18.89962B_i + 3347 \end{aligned} \tag{3.12}$$

If the energy of the current frame is greater than all the three thresholds, then Rate 1 is chosen. If it is greater than only two of these thresholds, then Rate 1/2 is chosen. If it is greater than only one threshold then Rate 1/4 is chosen. Otherwise, Rate 1/8 is chosen. However, the speech codec does not reduce the rate by more than one step at a time; the

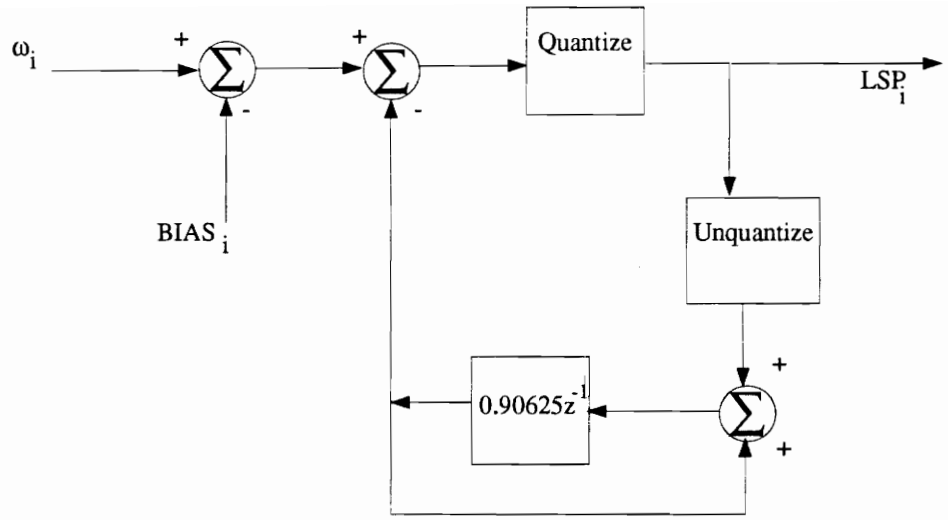

Figure 3.6: The quantization of the LSP differences

current rate is never less than half the previous rate. There is no stipulation on the amount of increase in the data rate.

3.2.7 Quantization of LSP frequencies

Each of the LSP frequencies are converted for transmission by subtracting from them a bias equal to

$$\text{Bias}_i = \frac{i}{2(M+1)} \qquad 1 \leq i \leq M$$

where $M = 10$ is the order of the LPC filter. Then the predicted value of the corresponding LSP frequency of the previous frame is subtracted from the current LSP frequency and the difference is quantized. This is analogous to DPCM. Instead of quantizing the difference in LSP frequencies of consecutive frames, the difference in adjacent LSP frequencies of the same frame could be quantized. The block diagram of this conversion and quantization operation is shown in Figure 3.6.

Each LSP frequency is quantized to 4 bits if the data rate is 9600 bps, 2 bits if it is 4800 bps, and 1 bit if it is either 2400 bps or 1200 bps. The maximum possible quantization level in either direction (positive or negative) for each LSP frequency according to the data rate is given in Table 3.3.

Table 3.3: LSP Quantization Levels

LSP Frequency	Rate 1	Rate 1/2	Rate 1/4	Rate 1/8
ω_1	0.025	0.015	0.01	0.01
ω_2	0.04	0.015	0.01	0.01
ω_3	0.07	0.03	0.01	0.01
ω_4	0.07	0.03	0.01	0.01
ω_5	0.06	0.03	0.01	0.01
ω_6	0.06	0.02	0.01	0.01
ω_7	0.05	0.02	0.01	0.01
ω_8	0.05	0.02	0.01	0.01
ω_9	0.04	0.02	0.01	0.01
ω_{10}	0.04	0.02	0.01	0.01

3.2.8 Interpolation of the LSPs

The quantized LSP differences are unquantized and converted back to the LSP frequencies. The stability of the LSPs are checked to make sure that the synthesis filter has not been rendered unstable due to quantization. If the LSPs are ordered as described in Section 3.2.5, then stability is ensured. Also, the LSPs are kept at least 80Hz apart. If they were too close, then there would be undesirable peaks in the LPC synthesis filter's response. (The minimum spacing translates to 80/(sample rate)=0.01 in the LSP scale).

Low Pass Filtering of the LSPs

The LSPs are passed through LPF to reduce quantization effects. These filtered LSPs $\{\hat{\omega}_i\}$ are given by

$$\hat{\omega}_i(\text{frame}) = x\hat{\omega}_i(\text{frame}) + (1-x)\omega_i(\text{frame}-1) \tag{3.13}$$

where x is a smoothing factor depending on the data rate. If the current data rate is Rate 1, then $x = 0$. If the data rate is Rate 1/2, then $x = 0.125$. If the data rate is either Rate 1/4 or Rate 1/8, then either $x = 0.125$ or $x = 0.9$, depending on whether there have been less than 10 consecutive frames of these rates or more than 10 respectively.

Interpolation

Since the LSPs were calculated from a speech window at the center of the last quarter of the frame, they must be interpolated to compute the corresponding values for each pitch subframe or codebook subframe within the frame. The interpolation is done based on the values for the current frame and the values for the previous frame. The coefficients of interpolation χ are dependent on the distance between the center of the current pitch subframe and the center of the windowed frame. For Rate 1, the values of χ are 0.75, 0.5, 0.25, and 0.0 for the four pitch subframes. For Rate 1/2, χ takes on values 0.625 and 0.125. For Rates 1/4 and 1/8, χ is 0.375. The interpolated LSPs are then computed as

$$\hat{\omega}_i(\text{frame}) = \chi\hat{\omega}_i(\text{frame} - 1) + (1 - \chi)\omega_i(\text{frame}) \tag{3.14}$$

The interpolated LSPs are then converted back to LPC coefficients by forming the polynomials $P(z)$ and $Q(z)$ using the LSPs multiplied by 2π as their roots. The odd LSP terms constitute $P(z)$ and the even terms constitute $Q(z)$. Finally the LPC coefficients are calculated as the average of the like powered coefficients of these two polynomials.

3.2.9 Pitch Parameters Determination

The pitch synthesis filter is expressed as

$$\frac{1}{P(z)} = \frac{1}{1 - bz^{-L}} \tag{3.15}$$

where b is the pitch gain, represented by three bits and a value from 0.0 to 2.0, and L is the pitch lag, represented by seven bits and a value from 17 to 143. Both b and L are determined for each pitch subframe.

An analysis-by-synthesis method is used to determine these two parameters. Those values for the parameters are chosen which minimize the weighted error between the input speech and the speech synthesized using these values. The codebook vector with all zero elements is used as input for synthesis. The pitch lag is varied between 17 and 143 in steps of one, and the pitch gain is varied from 0.0 to 2.0 in steps of 0.25.

Computation Procedure

The zero-input-response (ZIR) of the formant filter $1/A(z)$ is subtracted from the input speech for the current subframe to get the residual error. The formant filter is initialized with memories remaining in the decoder's formant filter. The residual error is then filtered by the perceptual weighting filter to get the perceptually weighted error. The perceptual weighting filter is given by

$$W(z) = \frac{A(z)}{A(z/\xi)}$$

where $A(z)$ is the LPC prediction error filter, and $\xi = 0.8$ is the perceptual weighting parameter.

The synthesis filter used in the encoder is given by

$$H(z) = \frac{1}{A(z/\xi)}$$

and is called the weighted synthesis filter because of the perceptual weighting parameter. The output of the pitch filter is estimated by filtering the current subframe of speech through the pitch filter for lag L and gain b. The filter has memories which correspond to the previous subframes. This output is then filtered by the weighted synthesis filter (without memory) to obtain the long-term prediction vector (the impulse response of $H(z)$ is truncated to 20 samples as its value is very small beyond this). The convolution is performed as follows.

$$y_L(n) = \sum_{i=0}^{\min(n,19)} h(i)p_L(n-i) \quad 16 < L \leq 143, \quad 0 \leq n < L_p$$

where $h(n)$ is the impulse response of $H(z)$, $p_L(n)$ is the output of the pitch filter, $y_L(n)$ is the zero-state-response (ZSR) of H(z), and L_p are the number of samples in each subframe. This is then subtracted from the perceptually weighted error obtained above to give the overall weighted error. The mean-squared value of the overall weighted error is minimized over all L and b.

The values of b and L are then quantized. The value $L = 16$ is used to signify the case where $b = 0.0$. The quantized values of b will be $PGAIN = b/0.25 - 1$ and of L will be set to $PLAG = L - 16$, if the gain is non-zero, else these are both set to zero.

3.2.10 Codebook Parameter Determination

Each pitch subframe consists of two codebook subframes except in the case of a Rate 1/8 packet. The codebook index I and the codebook gain G is determined for every codebook subframe. In case of a rate 1/8 packet, only one codebook subframe is present, and the index too is discarded after transmission.

The excitation codebook contains of 2^M predefined code vectors, where $M = 7$. The codebook is designed in a recursive fashion such that each code vector differs from the adjacent code vector by one sample. The samples in adjacent code vectors are shifted by one position such that a new sample is shifted in at one end and a sample is dropped at the other. A circular codebook of 2^M samples is used to simplify implementation and save memory.

Once again, an analysis-by-synthesis method is used to determine the codebook parameters. The chosen index I and gain G are those which minimize the weighted error between the synthesized speech and input speech. The synthesized speech is the codebook vector filtered by the pitch synthesis filter and the formant synthesis filter. The weighting filter has the same form as in the case of pitch computations. The LPC coefficients used for the filters in the current codebook search are those obtained from the interpolated LSPs for the current pitch subframe.

The search process for the codebook index is similar to that in pitch search. The ZIR of the pitch synthesis filter (with memory corresponding to the previous codebook subframe; lag and gain being those determined in the pitch search) is filtered by the formant synthesis filter $1/A(z)$ (with no memory). This output is subtracted from the speech corresponding to the current codebook subframe and the error is passed through the weighting filter $W(z)$. Then a code vector is selected from the codebook and scaled by the gain parameter (which can take eight possible values). This vector is then filtered by the weighted synthesis filter $H(z)$ to give the ZSR which is then subtracted from the perceptually weighted error to give the overall error whose mean-squared value is minimized over all possible code indices and gains. The convolution procedure to obtain the ZSR is the same as in the pitch search, except that the speech length is L_c, which is the length of the codebook subframe.

Table 3.4: Codebook Gain Prediction Filter

x	F_G(x)	x	F_G(x)	x	F_G(x)	x	F_G(x)
-6	-2	13	12	32	28	51	45
-5	-2	14	13	33	29	52	46
-4	-2	15	14	34	30	53	47
-3	-2	16	15	35	31	54	48
-2	-1	17	16	36	32	55	49
-1	0	18	17	37	33	56	50
0	0	19	18	38	34	57	51
1	0	20	18	39	35	58	52
2	1	21	18	40	36	59	53
3	2	22	19	41	36	60	54
4	3	23	20	42	37	61	54
5	4	24	21	43	38	62	55
6	5	25	22	44	39	63	56
7	6	26	23	45	40	64	57
8	7	27	24	46	41	65	58
9	8	28	25	47	42	66	58
10	9	29	26	48	43		
11	10	30	27	49	44		
12	11	31	27	50	45		

Conversion of Codebook Parameters into Transmission Codes

The index I is quantized to seven bits. If the gain G is negative, then the value $(I + 89)$ mod 128 is quantized and transmitted instead of I.

The gain G is converted to a dB scale by taking 20 times the log of the magnitude of G. Then it is coded using a differential coder which uses a 2-bit linear quantizer Q_G and a prediction filter $P_G(z)$. This differential coder operates on all of the codebook subframes for all of the rates.

The predictor is defined as

$$P_G(z) = F_G\left(\left\lfloor \frac{z^{-1} + z^{-2}}{2} \right\rfloor\right)$$

where $\lfloor . \rfloor$ indicates truncation to integer precision.

Table 3.4 shows the values F_G takes for different values of its argument. The codebook quantizer $Q_G(x)$ operates on the difference between the dB gain and the output of the prediction filter. For rates 1 and 1/2, the dependence of the output of the quantizer $Q_G(x)$

on the input x is as follows

$$\begin{aligned} x < -2 \quad , \quad Q_G(x) &= -4 \\ -2 \leq x < 2 \quad , \quad Q_G(x) &= 0 \\ 2 \leq x < 6 \quad , \quad Q_G(x) &= 4 \\ 6 \leq x \quad , \quad Q_G(x) &= 8 \end{aligned}$$

For Rates 1/4 and 1/8, the output is

$$\begin{aligned} x < -3 \quad Q_G(x) &= -4 \\ -3 \leq x < -1 \quad Q_G(x) &= -2 \\ -1 \leq x < 1 \quad Q_G(x) &= 0 \\ 1 \leq x \quad Q_G(x) &= 2 \end{aligned}$$

The transmitted gain parameter, $CBGAIN$ is set equal to 0, 1, 2, 3 in the above cases respectively if the gain is positive, otherwise 4, 5, 6 or 7 is sent. Hence three bits are necessary for gain quantization (except in the case of Rate 1/8 where the sign is always kept positive and hence only 0, 1, 2 or 3 are the values sent, for which only two bits are needed).

For rate 1/8 frames, the Gaussian random codebook is replaced by a pseudorandom code vector in the decoding section of the encoder and the decoder. The codebook index and the sign of the gain are not transmitted. The pseudorandom code vector is generated by a pseudorandom number generator which is the same both at the encoder and the decoder. This is achieved by using the transmitted 16-bit packet at Rate 1/8 as the seed for the generator at both ends. To ensure randomness of this packet, four pseudorandom bits are put into $CBSEED$, the transmitted seed.

The third, seventh, eleventh and fifteenth bits of a value SD, which is computed as follows, are the four bits of $CBSEED$.

$$SD(\text{new}) = (521SD(\text{old}) + 259) \bmod 65536$$

If a rate 1/8 packet with all ones is encountered, in order that it is not wrongly interpreted as null traffic data, a new $CBSEED$ is generated until it is not all ones. The packet is then repacked with this new $CBSEED$.

3.2.11 Packing

Finally, before the data is packed, parity check bits are generated for rate 1 packets. 10 parity check bits are generated from 18 bits using the BCH(28,18) code. These 18 bits are the most perceptually significant bits. These are $LSP_1[3]$, $LSP_2[3]$, $LSP_3[3]$, $LSP_4[3]$, $LSP_5[3]$, $LSP_6[3]$, $LSP_7[3]$, $LSP_8[3]$, $LSP_9[3]$, $LSP_10[3]$ among the LSPs. $CBGAIN_1[1]$, $CBGAIN_2[1]$, $CBGAIN_3[1]$, $CBGAIN_4[1]$, $CBGAIN_5[1]$, $CBGAIN_6[1]$, $CBGAIN_7[1]$, $CBGAIN_8[1]$ are the ones among the codebook gains. Here $LSP_i[3]$ is the MSB of LSP_i and $CBGAIN_i[1]$ is the second MSB of $CBGAIN_i$.

The generator polynomial of the BCH(28,18) code is

$$g(x) = x^{10} + x^9 + x^8 + x^6 + x^5 + x^3 + 1$$

Then a parity check bit which is a parity bit of the 28 bit codeword is computed by taking the XOR of all the 28 bits. Finally, all the data is packed into a 171 bit packet for rate 1, an 80 bit packet for rate 1/2, a 40 bit packet for rate 1/4, and a 16 bit packet for rate 1/8.

3.2.12 Decoding at the encoder

After each codebook subframe, a version of the decoder is run at the transmit side to update the memories of the different filters. A 160-element long uniform pseudorandom sequence of variance 0.629 (since the codebook has variance 0.629) is generated using the packet as seed for rate 1/8 (for other rates, the index I is used to get the code vector from the codebook). The code sequence thus obtained is scaled by the gain and passed through the pitch synthesis filter and the formant synthesis filter to update their memories. The output of the formant synthesis filter is also used to update the memories of the perceptual weighting filter, the output of which may be discarded. The parameters used for the filters are derived from their packed values.

The gain used for scaling the code vector in case of rate 1/8 is calculated after low-pass filtering (to prevent burstiness in the noise effects), and interpolating the gain value obtained after converting back from the transmission codes. The low-pass filtering action is achieved by averaging the gains for the previous and current codebook subframes. The interpolating factors are 0.875, 0.750, 0.625, 0.5, 0.375, 0.25, 0.125 and 0.0 for each eighth

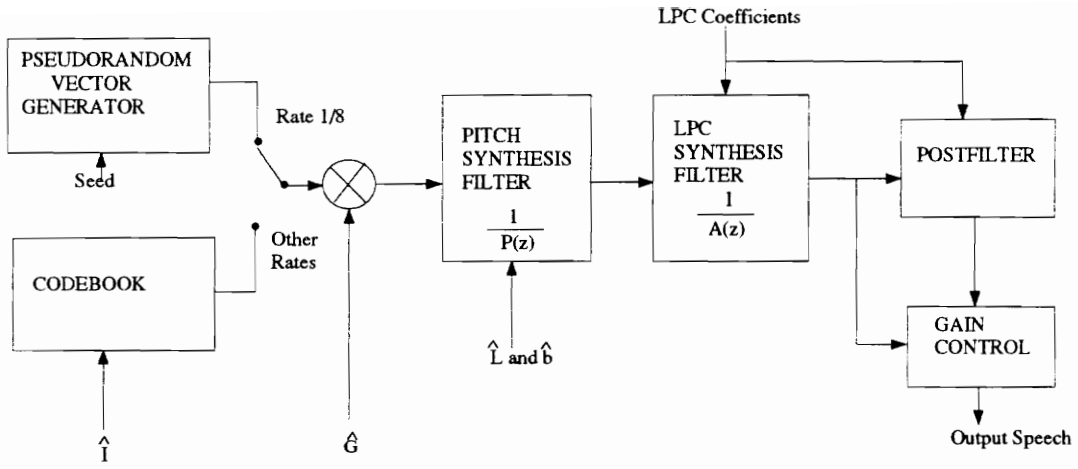

Figure 3.7: The Decoder Model

of a frame respectively. In case of other rates, the gain is directly used after converting back from the transmission codes.

3.3 Decoder Model

The block diagram for the synthesis of speech using the QCELP technique is shown in Figure 3.7. The scaled codebook vector is generated from the codebook index $\hat{I}$ and codebook gain $\hat{G}$ received from the encoder, as in Section 3.2.12. Then the pitch gain $\hat{b}$ and lag $\hat{L}$ are obtained from the packet after unquantizing them. The LSPs are also obtained from the packet and interpolated for the current pitch subframe as in Section 3.14. These interpolated LSPs are then converted to LPC coefficients. Then the scaled codebook vector is run through the pitch synthesis filter and the formant synthesis filter, the output of which is passed through the adaptive postfilter which has the form

$$PF(z) = B(z)\frac{A(z/p)}{A(z/s)}$$

where $A(z)$ is the formant prediction error filter, $p = 0.5$, $s = 0.8$. $B(z)$ is an anti-tilt filter designed to offset the spectral tilt introduced by $A(z/p)/A(z/s)$, and is given by,

$$B(z) = \frac{1 - \gamma z^{-1}}{1 + \gamma z^{-1}}$$

where γ is 0.25 if the average of the ten interpolated LSPs is not more than 0.24, -0.25 if it is more than 0.26 and is 25 times the difference $[[0.25 - \text{average}(interpolatedLSPs)]$.

The filter $PF(z)$ is initialized with memories resulting from the last output sample. The pitch synthesis filter operates at frequencies from 50 Hz to 800 Hz. The postfilter operates at frequencies from 50 Hz to 4 kHz. A gain control is applied, as seen in Figure 3.7 to ensure that the energy of the output of $PF(z)$ is roughly the same as the energy of the input signal. An initial scale factor is determined by computing the square root of the ratio of the energy in 40 samples of the input to $PF(z)$ to the energy in the 40 samples of the output of $PF(z)$. This scale factor is then filtered by adding 0.9375 times the final scaling value for the previous 40 samples to the initial scaling value for the current 40 samples. The final output is then the reconstructed speech which is computed by multiplying this scale factor with the output of the postfilter.

3.4 Forward and Reverse Channel Models in the IS-95 standard

The forward channel in a wireless system is the base-station to mobile link, whereas the reverse channel is the mobile to base-station to mobile link. This algorithm was written paricularly for the IS-95 standard [8] for wideband CDMA cellular telephony. The IS-95 system itself was implemented by Yingjie Li [28]. Therefore, it would be useful to provide an insight into the forward and reverse channel structures for this standard, wherein the speech coder implementing this algorithm is used at the input. Figures 3.8 and 3.9 show the block diagram for the reverse and forward CDMA channel structures respectively.

As seen in Figure 3.8, the information bits coming out from the vocoder at one of the four rates are encoded by a rate 1/3, constraint length 9, convolutional encoder. The output bits of the encoder are termed as code symbols, which are repeated either 8, 4, or 2 times for rate 1/8, rate 1/4, rate 1/2 respectively. These symbols are interleaved to combat the effects of bursty errors. An interleaver spanning 20 ms is used, with the interleaver matrix having dimensions 32×18. The interleaving action is performed by writing into the matrix columnwise and reading out rowwise. The code symbols output from the interleaver are then modulated using 64-ary orthogonal modulation. One of 64 possible modulation symbols is transmitted for every six code symbols, the index of the modulation symbol being the decimal equivalent of these 6 code symbols. The modulation symbol is one of 64 mutually orthogonal waveforms generated using Walsh functions.

Then, this output stream is gated with a time filter that allows transmission of certain

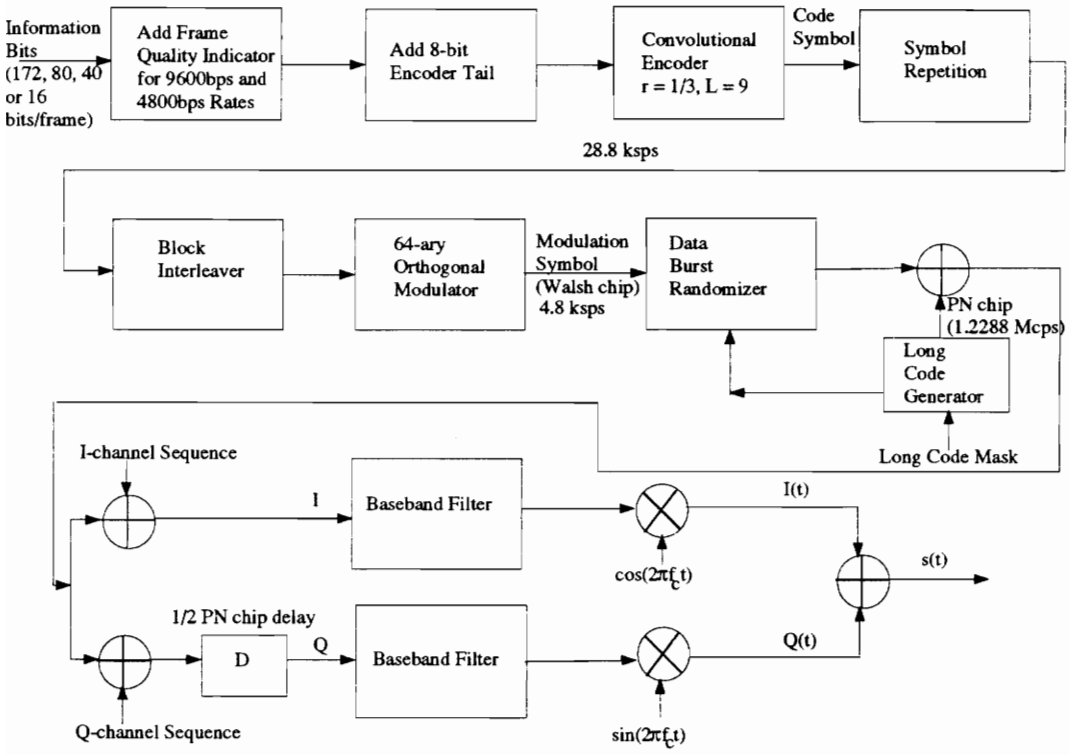

Figure 3.8: The Reverse CDMA Channel Structure

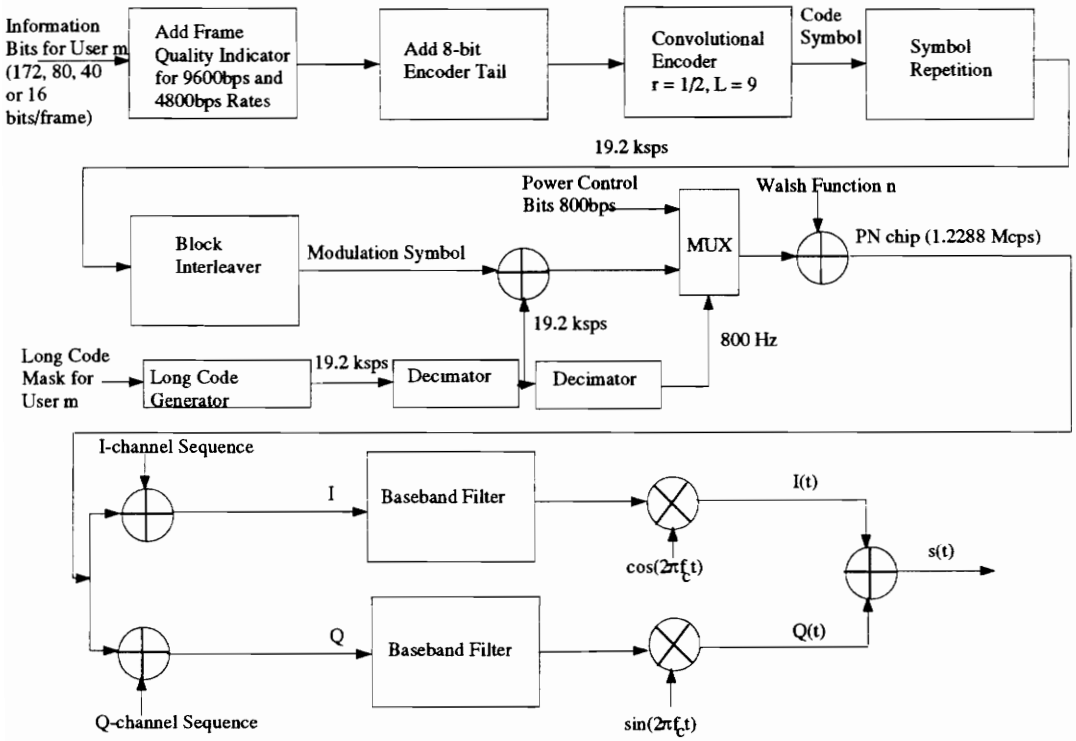

Figure 3.9: The Forward CDMA Channel Structure

interleaver output symbols and deletion of others. The duty cycle of the transmission gate varies with the data rate. When the data rate is 9600 bps, the transmission gate allows all interleaver output symbols to be transmitted. When the data rate is 4800 bps, 2400 bps or 1200 bps, the gate allows only 1/2, 1/4 or 1/8 of the interleaver output symbols to be transmitted. The gating process operates by dividing the frame into 16 groups, called power control groups. Certain power control groups are gated on while others are gated off. The assignment of gated on and gated off groups is referred to as the data burst randomizing function. The data burst randomizer ensures that every code symbol subject to the repetition done previously is transmitted exactly once. The data burst randomizer generates a masking pattern that randomly masks out the redundant data generated by the code repetition. This is determined by the data rate of the frame and the last 14 bits of the long code used for spreading in the power control group previous to the last in the previous frame. During the gated off periods, the mobile station reduces its average output power by at least 20dB from the average output power of the most recent power control group or to the transmitter noise floor, whichever is greater. As a consequence of this, the interference to other mobile stations on the same reverse channel is reduced.

The output of the data burst randomizer is then direct sequence spread by a long code which has a period of $2^{42} - 1$ chips. The chip rate is 1.2288 Mcps. The spreading is achieved by performing modulo-2 addition of the input to the spreader and the long code. Each Walsh chip (modulation symbol) is spread by 4 long code chips. Each PN chip of the long code is generated by the modulo-2 inner product of the 42-bit state vector and a 42-bit mask. The 42-bit mask has less significant 32 bits as the permuted bits of the Electronic Sequence Number (ESN). (The permutation is done to prevent high correlations between long codes corresponding to consecutive ESNs).

Following this, the reverse channel signal is quadrature spread using two PN sequences of the same chip rate as the long code. These are called the I-sequence and Q-sequence. These codes have a period of 2^{15} chips. The data spread by the Q-sequence is delayed by half a PN chip with respect to the data spread by the I-sequence. Then baseband filtering is done for the purpose of pulse shaping, after which the I and Q signals are modulated with a carrier.

From Figure 3.9, it is seen that the forward channel transmission process is somewhat

similar to the above, except for a few differences. The convolutional encoder is of rate 1/2 instead of rate 1/3. The interleaver matrix is of size 24×16. The output bits of the interleaver (modulation symbols) are then scrambled at 19.2 ksps rate. The data scrambling is achieved by performing the modulo-2 addition of the interleaver output symbol with the first long code PN chip of every 64 chips. (This PN chip selection constitutes the decimation action shown in Figure 3.9). The long code is generated at the rate of 1.2288 Mcps.

A power control subchannel is transmitted on the forward channel at the rate of one bit every 1.25 ms. A '0' bit is used to indicate to the mobile station to increase the average output power and a '1' bit to indicate a decrease in the average power level. The base station estimates the received signal strength of a particular mobile station over a 1.25 ms period (duration of a power control group of the reverse channel). Then the base station decides whether to send a 0 or a 1. This bit, which occupies a duration equal to two modulation symbols, replaces two consecutive modulation symbols before transmission of the forward channel signal.

Each code channel (a channel containing one user's information) is spread with a Walsh function at a chip rate of 1.2288 Mcps to provide orthogonality among all code channels on a given forward channel. One of 64 time-orthognal Walsh functions will be used for spreading a code channel. The code channel spread by Walsh function n is assigned the code channel number n ($n = 0$ to 63). The Walsh spreading sequence has a period of 64 chips.

Following orthogonal spreading, each code channel is spread in quadrature using two PN sequences, the I-sequence and the Q-sequence, each of length 2^{15} chips. Then the I and Q signals are baseband filtered as in the reverse channel.

Chapter 4

Simulation of QCELP

4.1 Introduction

In this chapter, the software implementation of the QCELP vocoder implemented in this thesis is described. The QCELP vocoder, channel models, and the IS-95 transmitter model were integrated into a simulation package suitable for evaluating subjective speech quality. The technical contribution of this thesis was significant programing work on the speech coder implementation and the integration of all system elements into a single package. Some portions of the QCELP vocoder – the calculation of line spectral pairs, the zero input response of the LPC synthesis filter and the pitch synthesis filter, were made available by Qualcomm. The IS-95 system and channel results are based on work by Li [28] and by Lichtenstein [29].

There are three different options which the simulation program of the voice coder offers. First the input can be a raw speech file and the output would be a packet file; in this case only the encoding part of the program is run. The second option is to give a packet file as input and obtain a raw speech file as output. Only the decoder runs in this case. Finally, the encoder and decoder can be placed in tandem to perform the vocoder's operation in one step, i. e. , the input raw file is converted to packets suitable for transmission which are passed on to the decoder to recover the speech in an output raw file.

4.2 Simulation Procedure

A block diagram of the simulation process is shown in Figure 4.1.

The input is obtained from files containing recorded speech. The speech was recorded by Bell Atlantic Mobile for use in testing the performance of vocoders. The test sequences used in this thesis were from one female and one male speaker. The female speech sequence was "the rabbits and dogs drowned; music can calm the nerves". This sequence was 4 seconds

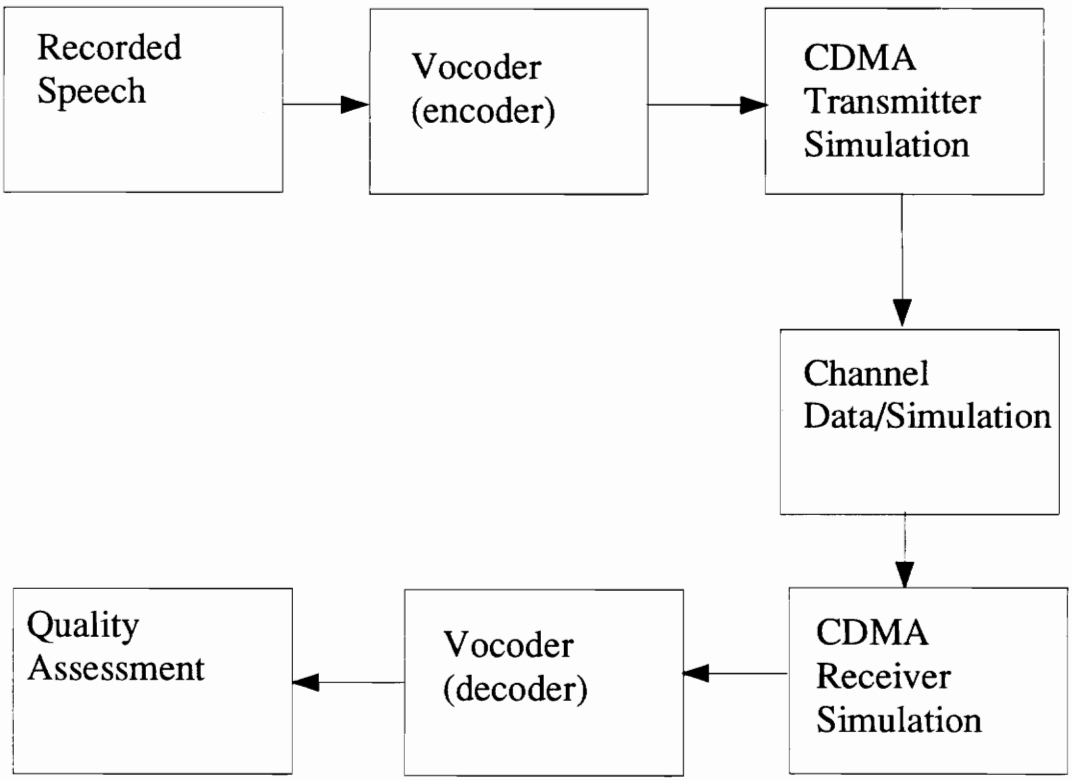

Figure 4.1: The Simulation Process

long. The male speech sequence was "they enjoy loud concerts; that flower is in bloom". This sequence was 5.25 seconds long. The input speech samples are then processed by the encoder of the vocoder implemented in this thesis. The data bits output from the encoder are then processed by the CDMA simulation implemented by Yingjie Li [28]. Multiple access interference was simulated by increasing the number of users to more than one. The CDMA simulation provided capability to specify the number of users. In this thesis, the speech was tested for 1, 10, 20 and 30 users. Since the CDMA simulation simulates the interfering users as though they had constant voice activity, these numbers of users translate to numbers closer to twice these values if all the other users also used variable rate vocoders such as that implemented in this thesis. The transmitted signal is convolved with the channel in the CDMA simulation itself and then the data bits are recovered by passing this signal through the CDMA receiver. The channel data is also provided by Bell Atlantic Mobile for both rural and urban environments. A rural environment has very few reflections of the transmitted signal and hence very few multipaths resulting in a mild fading effect. An urban environment on the other hand has many multipaths contributing to the fading and thus causing the transmitted signal to be corrupted to a greater extent than it would have been in a rural environment. A Rayleigh fading environment which is due to fading caused by the motion of the speaker is also simulated. A 1 Ray Rayleigh fading channel is simulated by multiplying the transmitted signal by a single envelope of a Rayleigh fading signal, whereas a 2 Ray Rayleigh fading channel is simulated by multiplying the transmitted signal by two Rayleigh fading envelopes with one being delayed with respect to the other by a random number of samples and with a lesser amplitude than the latter. The recovered data bits are then passed through the decoder of the vocoder implemented in this thesis to get back the speech transmitted. This received speech is then subjected to quality assessment based both on subjective and objective measures. The subjective methods involved having listeners rate the speech on a 5-point scale to obtain the Mean Opinion Score discussed in Section 2.4. The objective methods used were signal-to-noise ratio values to measure the distortion in the output speech relative to the input speech.

Figures 4.2 and 4.3 show the input speech for female and male voice respectively. These were the test speech sequences used throughout in the simulations.

The simulation proceeds by reading the speech file into an array using the MATLAB

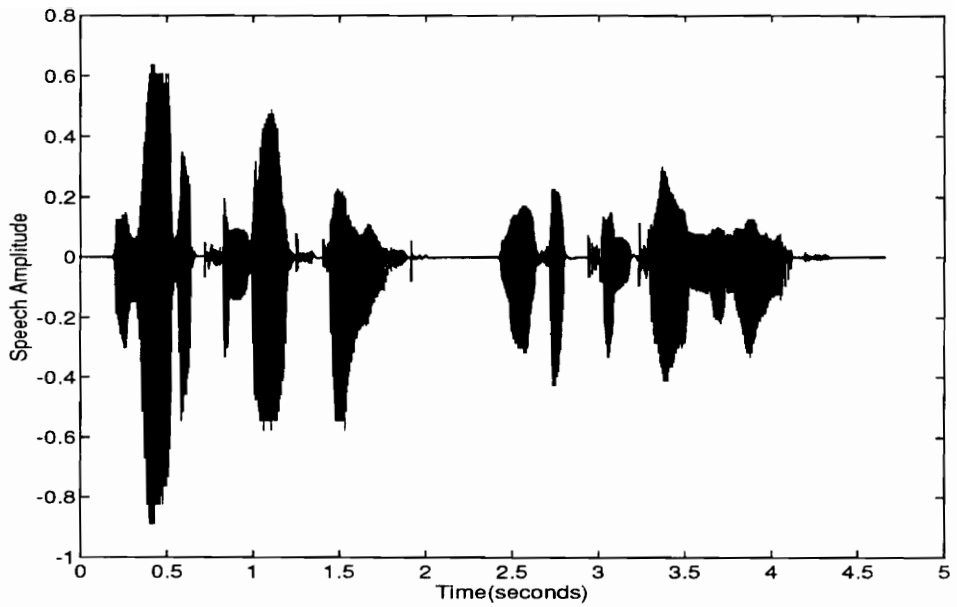

Figure 4.2: Input Female Speech to the Vocoder

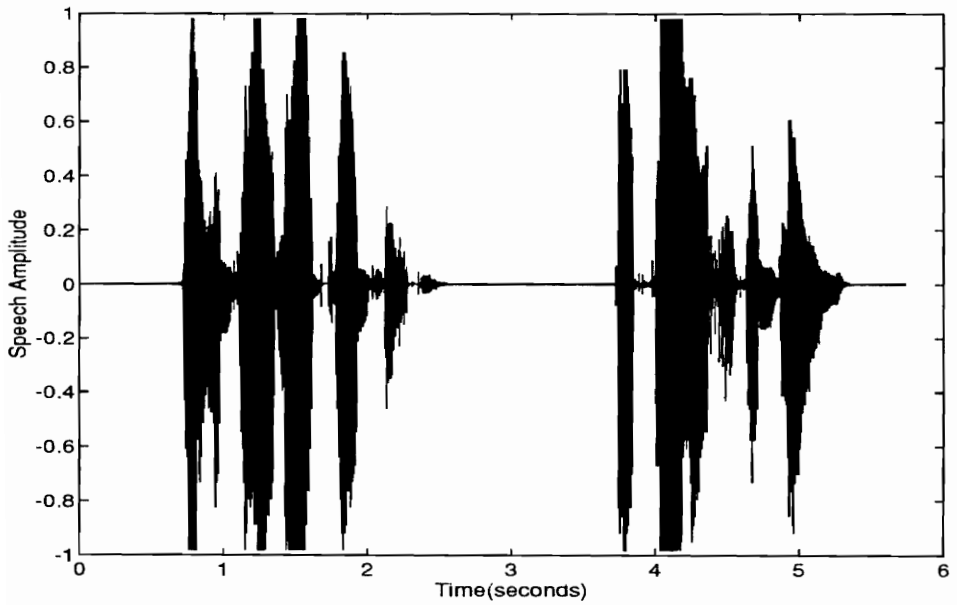

Figure 4.3: Input Male Speech to the Vocoder

function **auread**. These samples are then converted into a raw data stream in the form of two bytes per sample, the first byte being the lower byte and the second being the upper byte. This achieves a quantizing of the speech samples into a range of -32768 to 32767. This quantization is done by the function **adc** in the file **adc1.c**. Both these functions are called in the main program **analysis.c**.

After these steps, the QCELP encoding operation is performed to generate a packet file. The packets in this file are then subjected to different channels by XORing the bit-by-bit error pattern generated by simulating the effect of these channels on a random bit stream. These patterns were obtained through the CDMA simulation described in [28]. The Rayleigh fading envelopes for the channel data in Figure 4.1 were obtained from a MATLAB script file **ray.m** which uses Smith's method to compute them. The receiver speed chosen was 100kmph.

The resulting packets which have some of their bits in error are then used to reconstruct the raw file by performing the QCELP decoding operation. The speech itself is reconstructed from this raw file by doing an operation opposite to the quantization performed in the beginning; which then is written into an audio file using the MATLAB function **auwrite**. This audio file is then processed by the program **raw2audio** to convert it into an audio file that could be played on SUN workstation speakers by adding some header information.

4.2.1 QCELP Encoding and Decoding Operations

The previous section described the simulation outline. In what follows, the encoding and decoding operations alluded to above are described in detail with respect to the role played by different files involved in these operations.

The file **qcelp.c** is the main file which calls the different functions to perform the encoding and/or decoding as desired. The executable **qcelp** takes three command line options and/or arguments. The first of them is for indicating the input file, the second for the output file and the third for indicating whether either encoding or decoding is to be done or both.

The usage is as follows:

qcelp -i *rawfile* **-o** *packetfile* **-e**

qcelp -i *packetfile* **-o** *rawfile* **-d**

qcelp -i *inputfile* **-o** *outputfile*

The first line is used to perform encoding, the second to perform decoding and the last is (actually a combination of the first two) to just pass the speech through the vocoder and observe the performance of the vocoder in the absence of any external channel.

The file **encode.c** contains the function **encoder** to perform the encoding operation. The speech is processed to remove the DC and then is windowed by a Hamming window. This windowed speech is then processed to obtain the LSP parameters as described in Section 3.2. The pitch lag and gain are then calculated for each pitch subframe and the codebook index and gain are calculated for each codebook subframe. The version of the decoder present at the encoder is then run to update the memories of all relevant filters, the LPC synthesis filter, the weighted synthesis filter and the perceptual weighting filter. The file **lpc.c** contains the functions which calculate the autocorrelation values and then the LPC coefficients themselves from the windowed speech. The file **lsp.c** contains the routines to calculate the LSP frequencies from the corresponding LPC coefficients.

The routines in the file **quantize.c** are then used to quantize the LSPs. This file also contains functions to quantize the pitch gain and lag, and also the codebook gain and index. There are functions which do the unquantizing operation also in this file.

The file **pitch_code.c** has functions to calculate the pitch period, pitch gain, codebook index and codebook gain.. This also has utilities to get the appropriate filtered sequences which are then used in the computations of the pitch gain and lag, and the codebook gain and index. The file **filter.c** has filtering routines which are used to find out the zero-input-response (ZIR), zero-state-response (ZSR), and response for all-zero filters, all-pole filters, and pole-zero filters.

All the parameters are then quantized and packed into packets. The packing is done in **pack.c**. This file contains routines for packing data and also unpacking data to get back the different parameters.

The file **decode.c** contains the decoding routines. First the data is unpacked according to the rate of the arriving packet and then the corresponding parameters are unquantized and recovered. The speech is then reconstructed by passing the excitation sequence weighted by the codebook gain through the pitch synthesis filter and the formant synthesis filter. Finally, the output of this filter is applied to the postfilter to obtain the speech output after

suitable gain control. The details of these operations are described in the previous chapter. It also houses the routine **dec_part** which performs the first two filtering operations. Hence, this routine is also called during encoding to update the synthesis filters' memories. The file **init.c** initializes the encoder and decoder.

The file for reading and writing data is **io.c** in which there are routines to read in either samples of speech or words from a packet, and to write either samples of speech output or words into the packet file.

In all of the simulations for introducing errors into the transmitted packet data the first two bytes are left alone, since they contain the rate information. In the real time system however, the decoder determines the rate by the timing information, i. e. , by determining the number of bits arriving in a period of time. This being a simulation where the time factor cannot be explicitly incorporated, the rate is also packed into the packet. The rate information has to be protected in this simulation as otherwise the decoder part of the vocoder would find it impossibly difficult to reconstruct the speech and the program would terminate. This is so because the decoder expects a packet of rate 1, rate 1/2, rate 1/4, rate 1/8, rate 1 with errors, or blank packets. It thus figures out one of these possibilities by checking the word in the packet corresponding to the rate. Hence, the byte containing the rate is left untouched. The protection is achieved by stripping the rate from the packets and subjecting the remaining part of the packet to the CDMA channel.

Figure 4.4 shows the speech output after processing by the vocoder for the input speech in Figure 4.2. When compared with the input speech in Figure 4.2, it is observed that the change in the waveforms is imperceptible.

Figures 4.5 and 4.6 show the error in output female speech for 1 user and 20 users for the forward channel in a rural environment. The error is computed by taking the difference between the input and output speech. It is observed that there is almost no distortion in case of one user. However, there is some perceivable amount of distortion when there are 20 users.

In this chapter the basic operation of the QCELP simulator was demonstrated. In the next chapter, the performance results are presented.

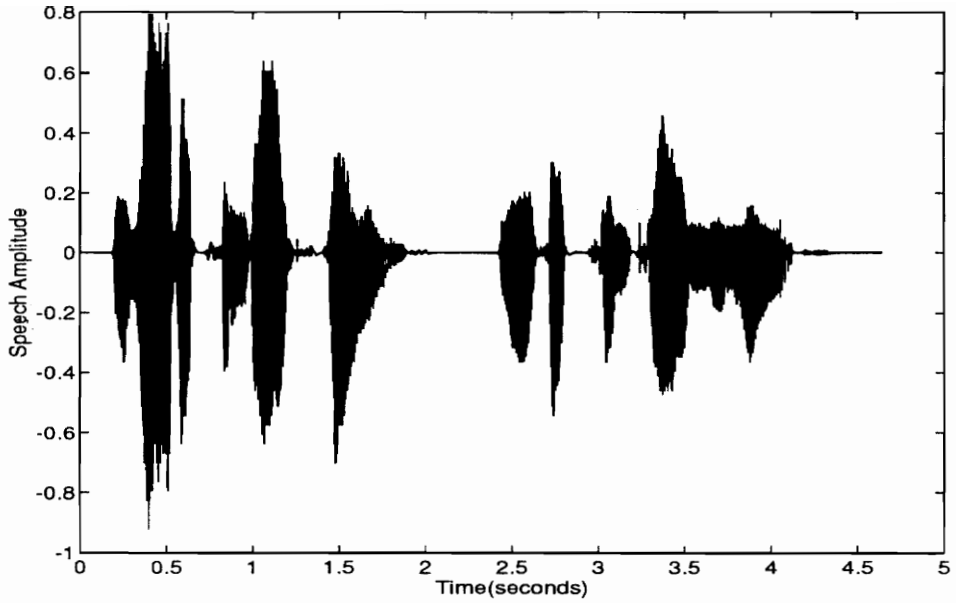

Figure 4.4: Output Female Speech from the Vocoder (no channel)

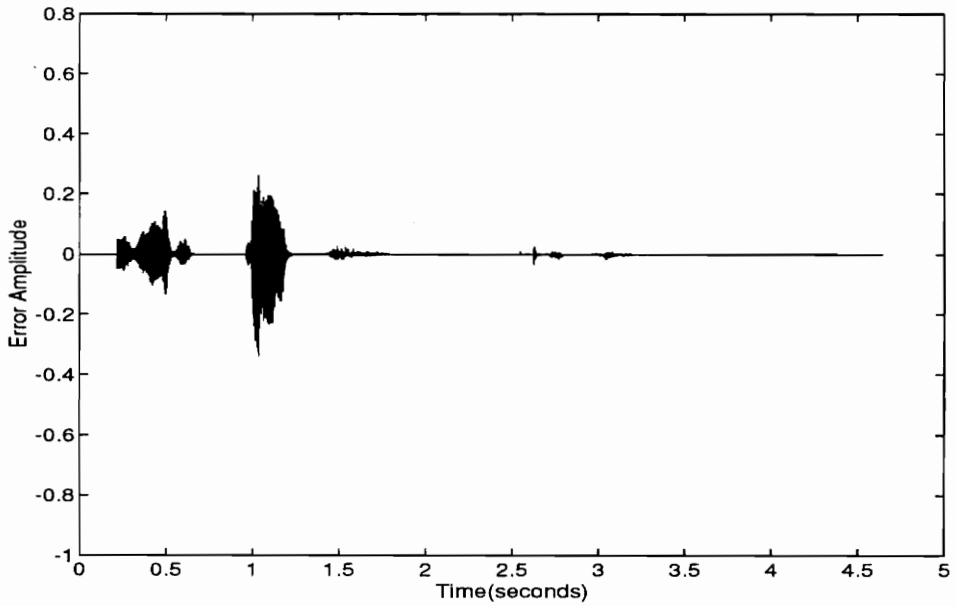

Figure 4.5: Error in Output Speech - Rural Environment, 1 User, 10dB SNR

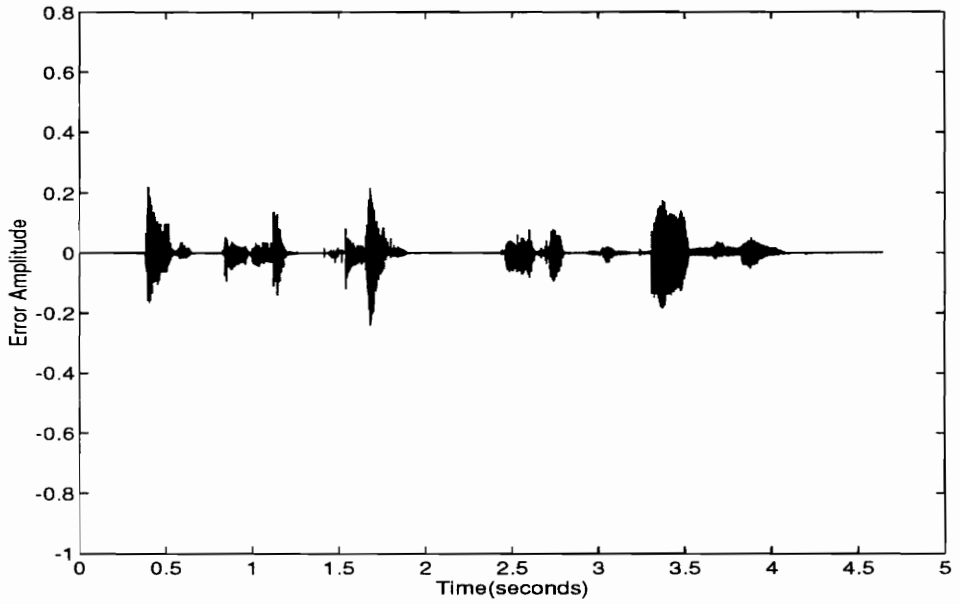

Figure 4.6: Error in Output Speech - Rural Environment, 20 Users, 10dB SNR

Chapter 5

Quality Assessment of QCELP

5.1 Introduction

In this chapter the performance of the QCELP vocoder under a variety of conditions is investigated. The input speech was obtained from recorded speech files provided by Bell Atlantic Mobile for both a male and a female speaker. The input speech was processed by the vocoder and subjected to the CDMA environment. Speech outputs were obtained for both rural and urban environments as well as for 1 Ray and 2 Ray Rayleigh fading environments. Both subjective and objective performance measures were considered. Mean Opinion Score (MOS) testing measures were used to determine the perceptual quality of the output speech. The results from this were then compared to the results from the objective measure, the Segmental SNR.

5.2 Subjective Measures

Mean Opinion Score testing was used as the subjective quality assessment measure in this thesis. MOS testing primarily involves collecting opinions from various listeners as to the quality of speech. The opinions take the form of a numerical grade on the speech outputs. The most popularly used grading scheme is the five-point scale. This scale consists of grades (scores) ranging from 5 down to 1. The relation between the scores and the speech quality is outlined as in Table 5.1 [17].

The opinion scores obtained by each listener are then averaged to get the mean score for each speech output.

The MOS measures are useful since they can accommodate listeners' freedom to decide what they consider as good or bad speech quality. Hence the test is applicable to a variety of distortions. However, listeners' opinions about good quality can vary greatly. Consistent test conditions have to be maintained; for instance, the order of presentation, environmental

Table 5.1: The Five-Point Scale for MOS Testing

Score	Speech Quality	Level of Distortion
5	Excellent	Imperceptible
4	Good	Just perceptible but not annoying
3	Fair	Perceptible and slightly annoying
2	Poor	Annoying but not objectionable
1	Unsatisfactory	Very annoying and objectionable

conditions for listening, etc.

It is essential to have test speech with which to unbias the listeners' opinions. In this particular testing, the input speech was first played to the listeners and this indicated to them as being excellent. The listeners were asked to base their opinions on this reference speech.

12 subjects listened to both the male and female speech outputs, obtained by subjecting the speech inputs to various channel conditions with various numbers of users in the CDMA system.

The results obtained are summarized in the following figures. Figures 5.1 and 5.2 show the MOS for female speech outputs for a rural environment at 10dB channel SNR for the forward and reverse channels respectively. The rural environment has very few reflections of the transmitted signal and hence the signal undergoes very little fading. As can be seen, the MOS drops as the number of users increases. Yet, the MOS for even 30 users was fair.

Figures 5.3 and 5.4 show the MOS for female speech outputs for an urban environment at 10dB channel SNR for the forward and reverse channels respectively. Again, it can be seen that the MOS drops as the number of users increases, as expected.

Figures 5.5 and 5.6 show the MOS for female speech outputs for urban environment at 5dB channel SNR. It is seen that as compared to the MOS for 10dB channel SNR, these scores are lower. This is because of the lower SNR of the channel.

Figure 5.7 compares the MOS for female speech outputs for both the channels in a rural environment at 10dB channel SNR. The MOS for the forward channel is expected to be better than that for the reverse channel because of the orthogonal spreading that takes place in the forward channel. This is observed to occur when there are large number of

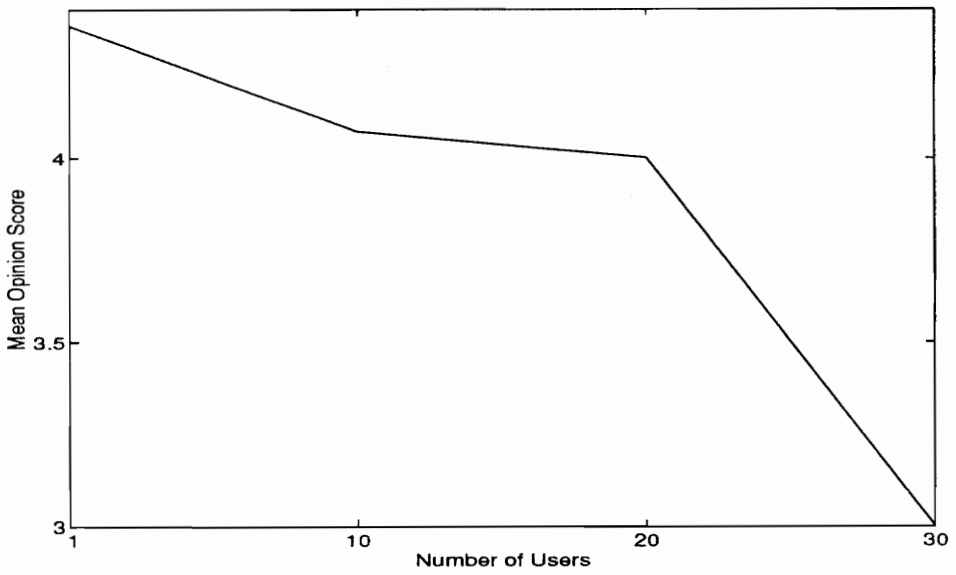

Figure 5.1: MOS - Female Speech, Forward Channel, Rural Environment, and 10dB channel SNR

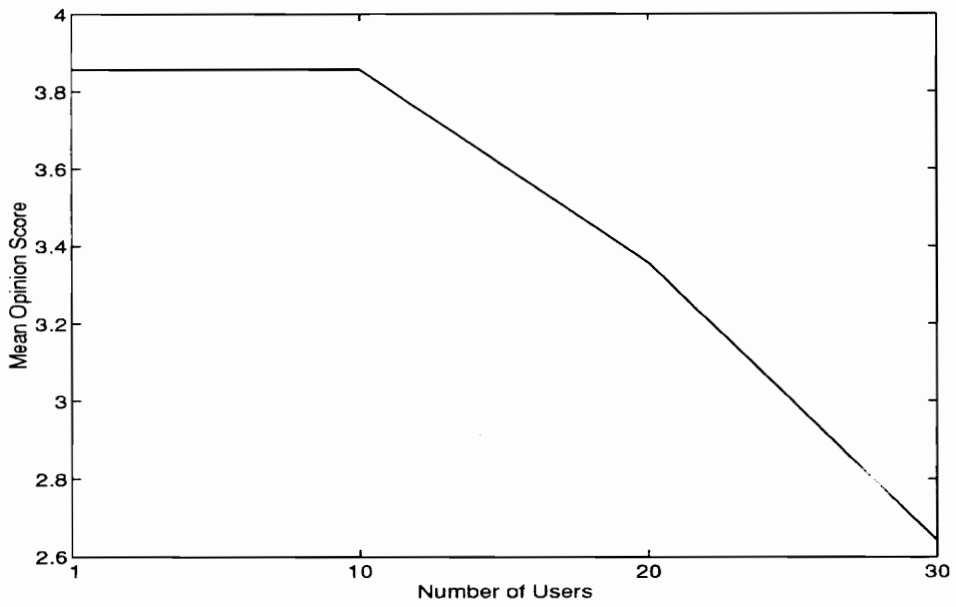

Figure 5.2: MOS - Female Speech, Reverse Channel, Rural Environment, and 10dB channel SNR

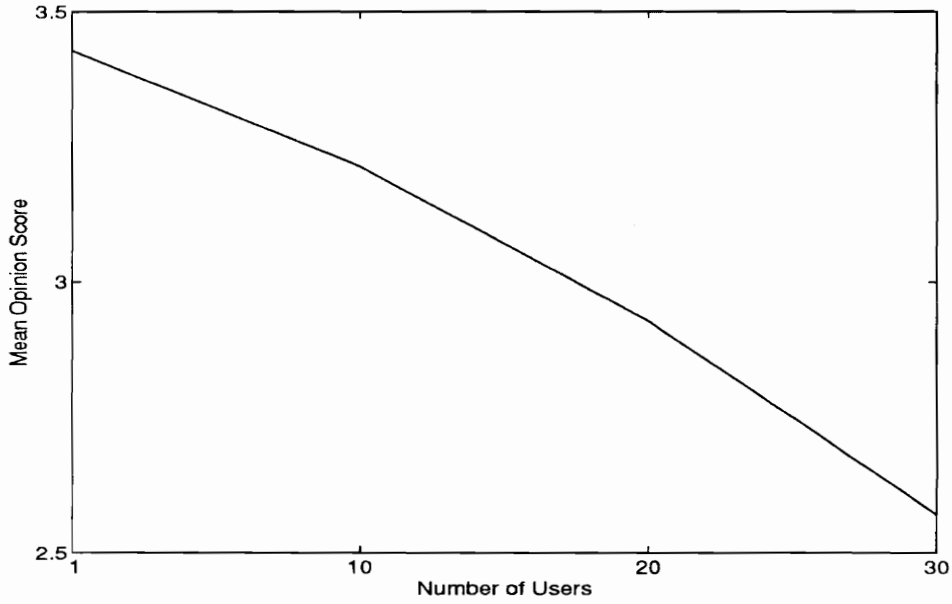

Figure 5.3: MOS - Female Speech, Forward Channel, Urban Environment, and 10dB channel SNR

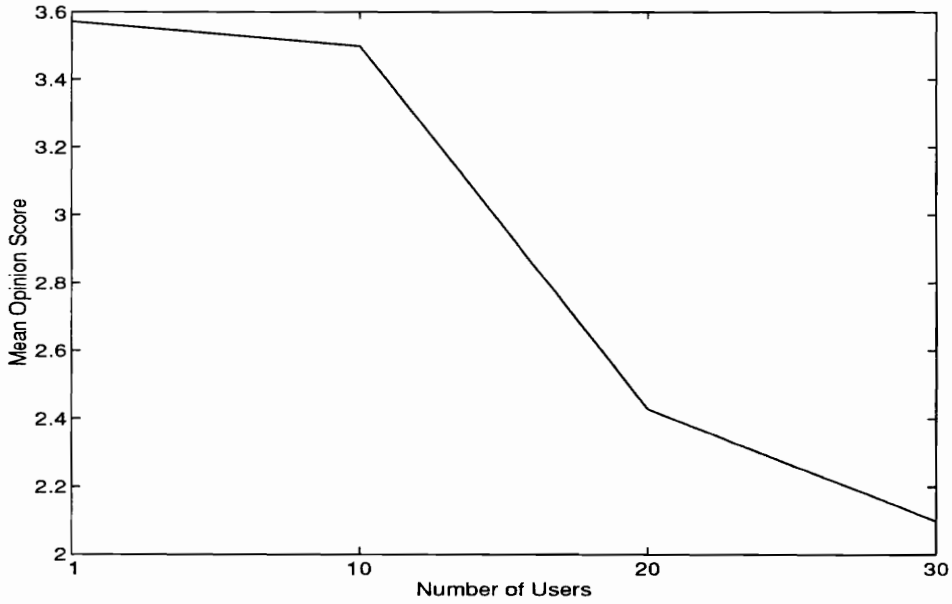

Figure 5.4: MOS - Female Speech, Reverse Channel, Urban Environment, and 10dB channel SNR

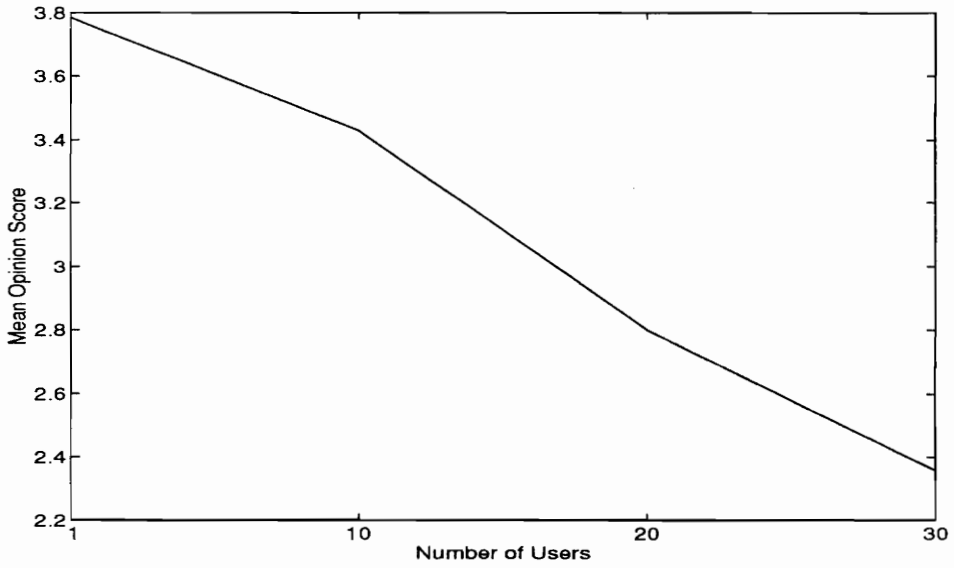

Figure 5.5: MOS - Female Speech, Forward Channel, Urban Environment, and 5dB channel SNR

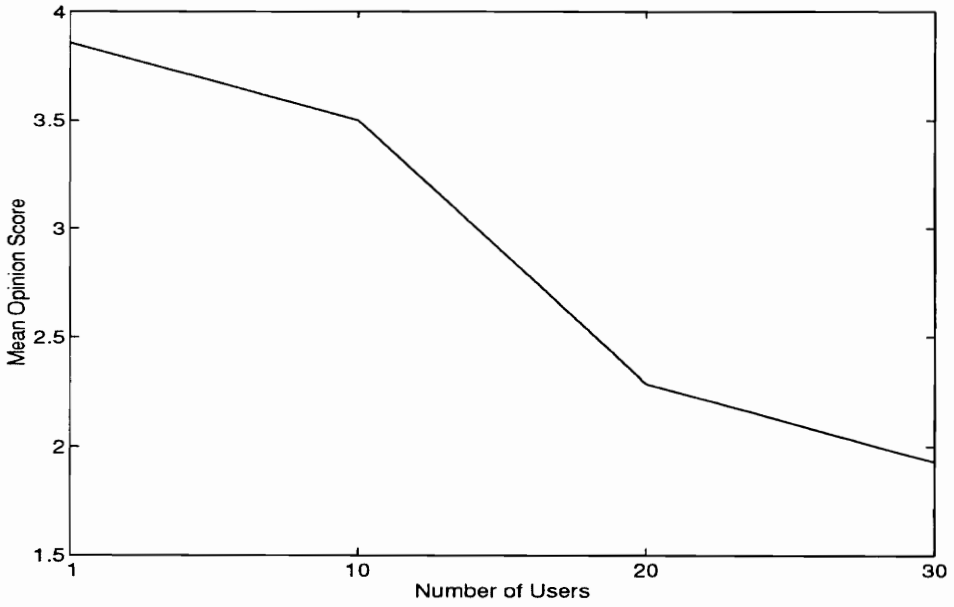

Figure 5.6: MOS - Female Speech, Reverse Channel, Urban Environment, and 5dB channel SNR

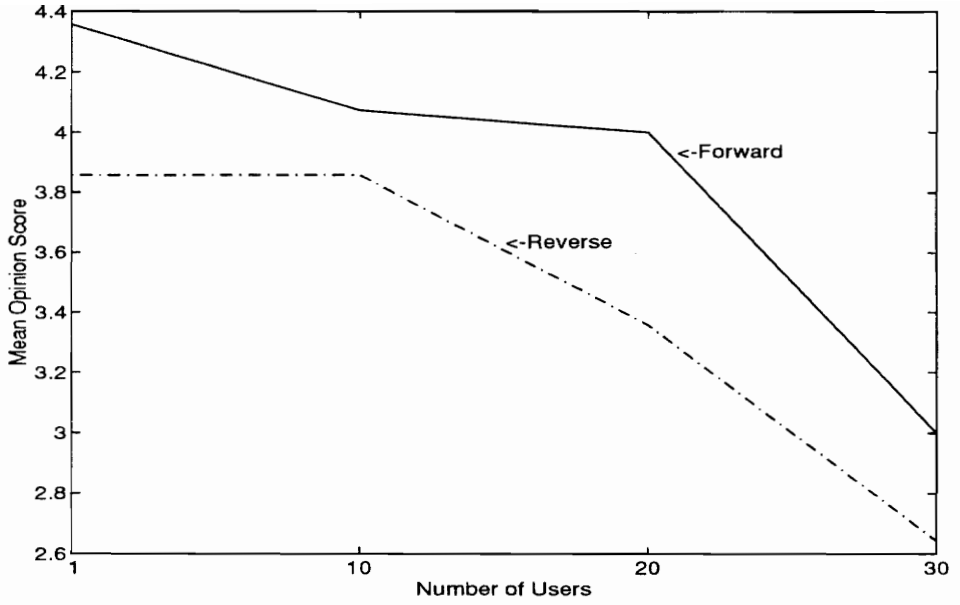

Figure 5.7: MOS comparison - Female Speech, Rural Environment, and 10dB channel SNR

users. When there are more users, the orthogonality of the signals in the forward channel becomes significant and provides large improvement over the performance in the reverse channel.

Figure 5.8 compares the MOS for female speech outputs for both the channels in an urban environment at 10dB channel SNR. Again, it is observed that, for a larger number of users, the forward channel has higher MOS than the reverse channel.

Figures 5.9 and 5.10 show the MOS for female speech outputs for 1 Ray Rayleigh Fading. These figures show the MOS for the 1 Ray Rayleigh Fading channel to be as good as the MOS in the rural environment. The interleaver-deinterleaver combination helps to combat the effects of this fading.

Figures 5.11 and 5.12 show the MOS for female speech outputs for 2 Ray Rayleigh Fading. The MOS here is observed to be less than that for 1 Ray Rayleigh Fading. The trend however, is the same; the MOS drops as the number of users increases.

Figures 5.13 and 5.14 show the MOS for male speech outputs for rural environment at 10dB channel SNR. The MOS is seen to drop as the number of users increases as in the case of female speech. However, the MOS is lower than that for the female speech. The reason

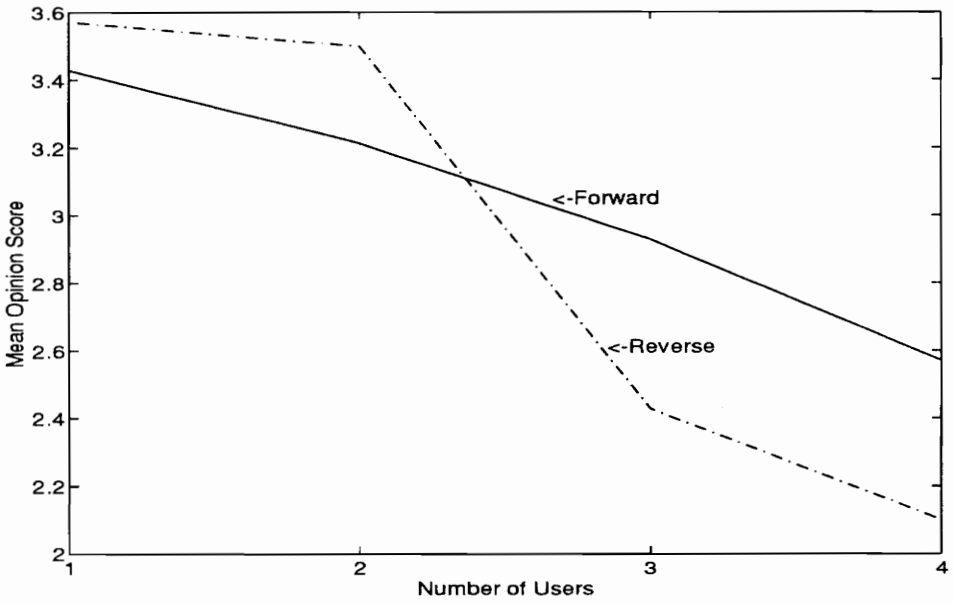

Figure 5.8: MOS comparison - Female Speech, Urban Environment, and 10dB channel SNR

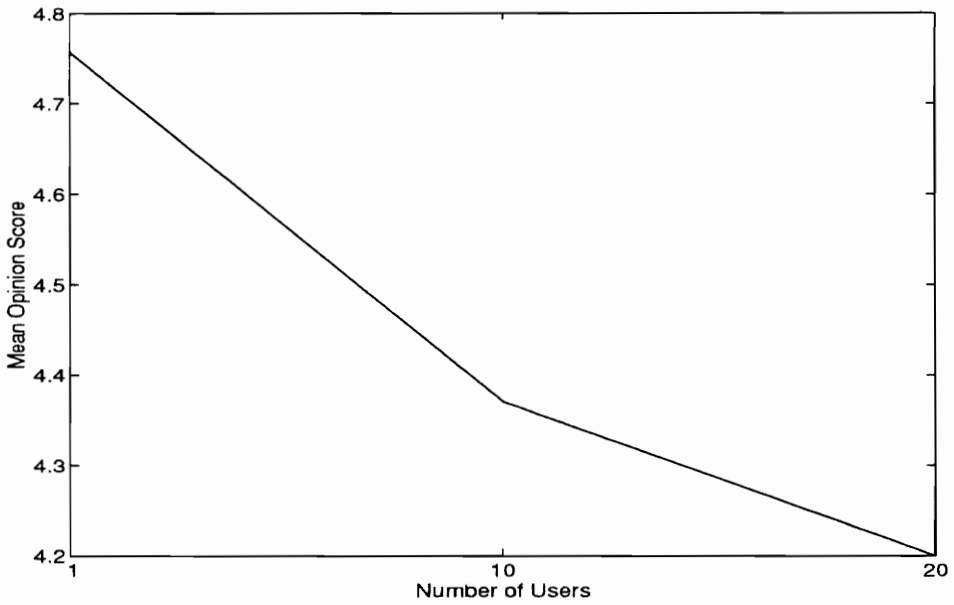

Figure 5.9: MOS - Female Speech, Forward Channel, 1 Ray Rayleigh Fading

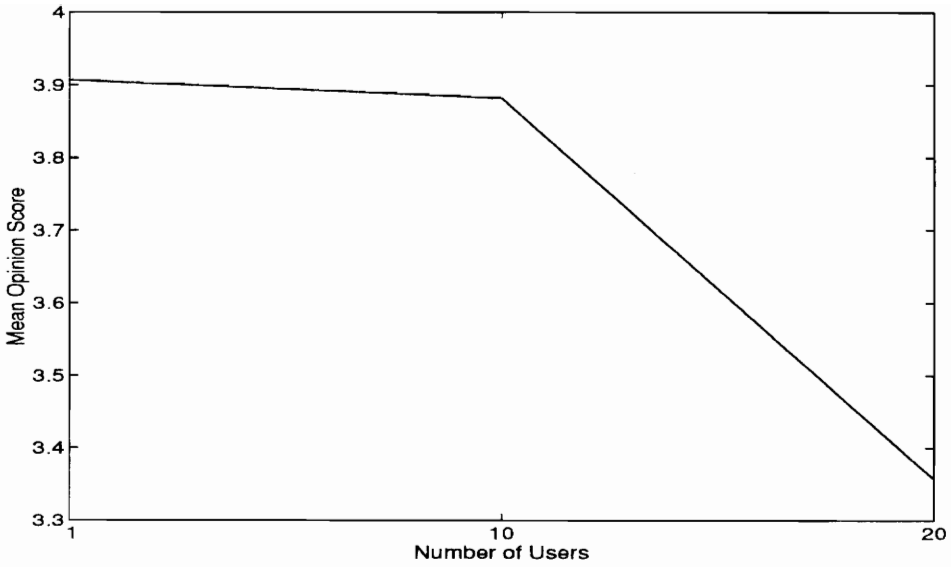

Figure 5.10: MOS - Female Speech, Reverse Channel, 1 Ray Rayleigh Fading

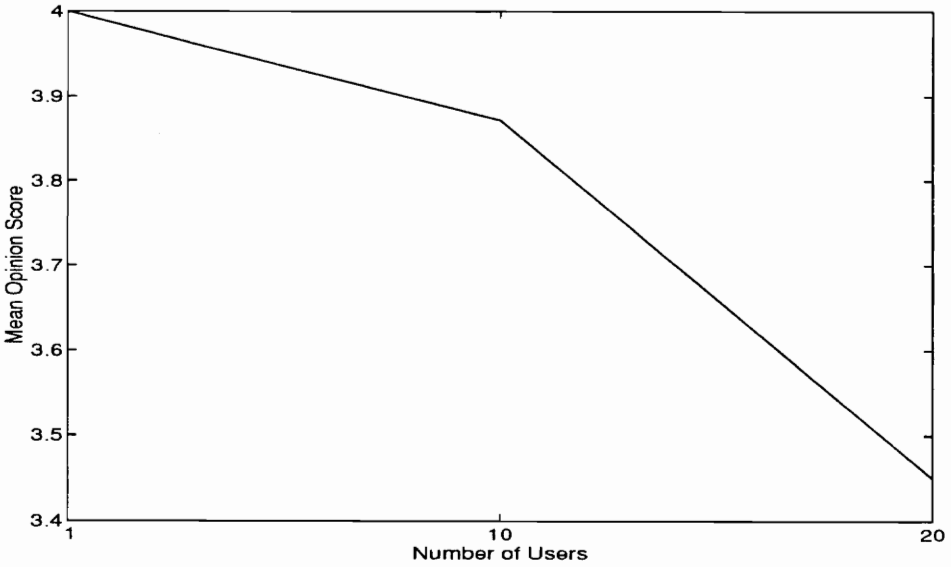

Figure 5.11: MOS - Female Speech, Forward Channel, 2 Ray Rayleigh Fading

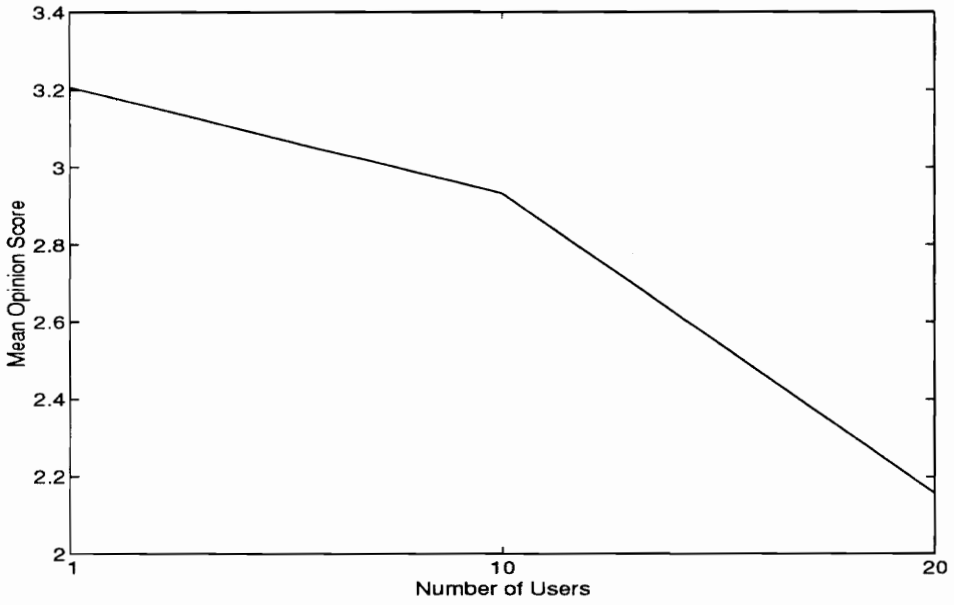

Figure 5.12: MOS - Female Speech, Reverse Channel, 2 Ray Rayleigh Fading

for this behavior was not obvious. It could probably be attributed to the male speech being less clearly spoken than the female speech.

Figures 5.15 and 5.16 show the MOS for male speech outputs for urban environment at 10dB channel SNR. Similar trend as in the case of female speech is observed. The results are worse, however. The male speech degrades a lot for large number of users in the urban environment.

Figures 5.17 and 5.18 show the MOS for male speech outputs for urban environment at 5dB channel SNR. As expected, the degradation is more marked than the case of 10dB channel SNR. Again, the speech quality falls as the number of users increases.

Figure 5.19 compares the MOS for male speech outputs for both the channels in a rural environment at 10dB channel SNR. Like the case of female speech, the forward channel performs better than the reverse channel for larger number of users, by virtue of the orthogonal spreading in the former as discussed in Section 3.4.

Figure 5.20 compares the MOS for male speech outputs for both the channels in an urban environment at 10dB channel SNR. The speech through the forward channel has better quality compared to the reverse channel.

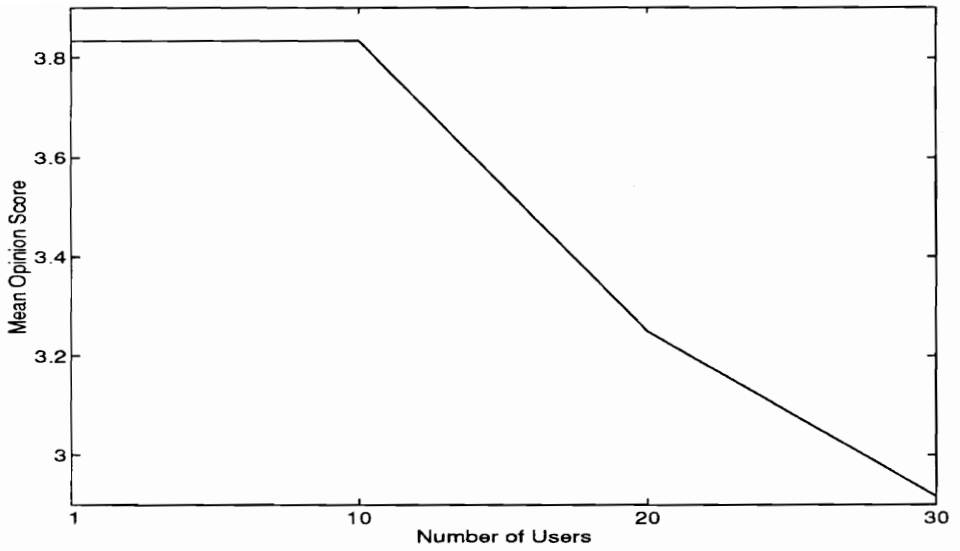

Figure 5.13: MOS - Male Speech, Forward Channel, Rural Environment, and 10dB channel SNR

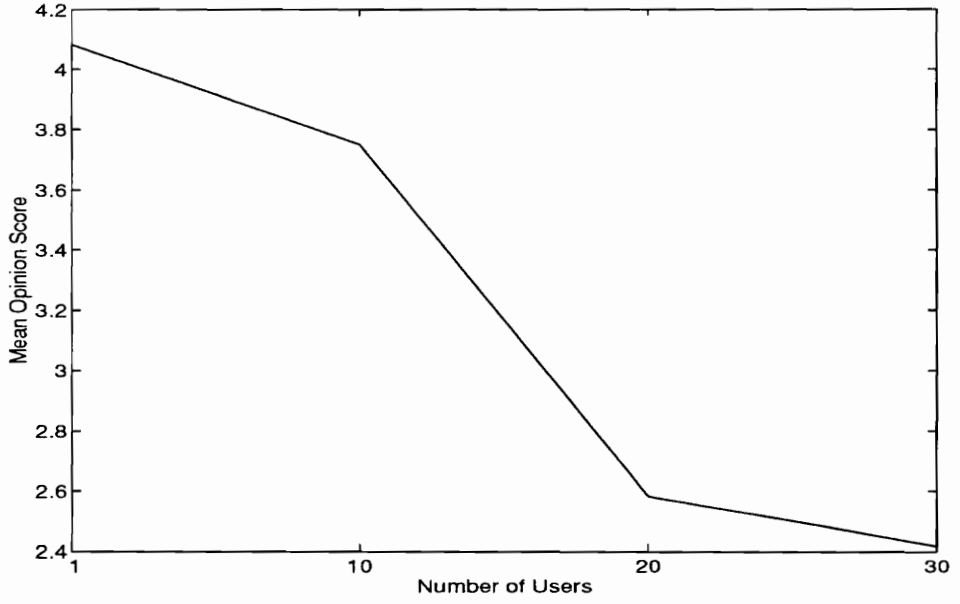

Figure 5.14: MOS - Male Speech, Reverse Channel, Rural Environment, and 10dB channel SNR

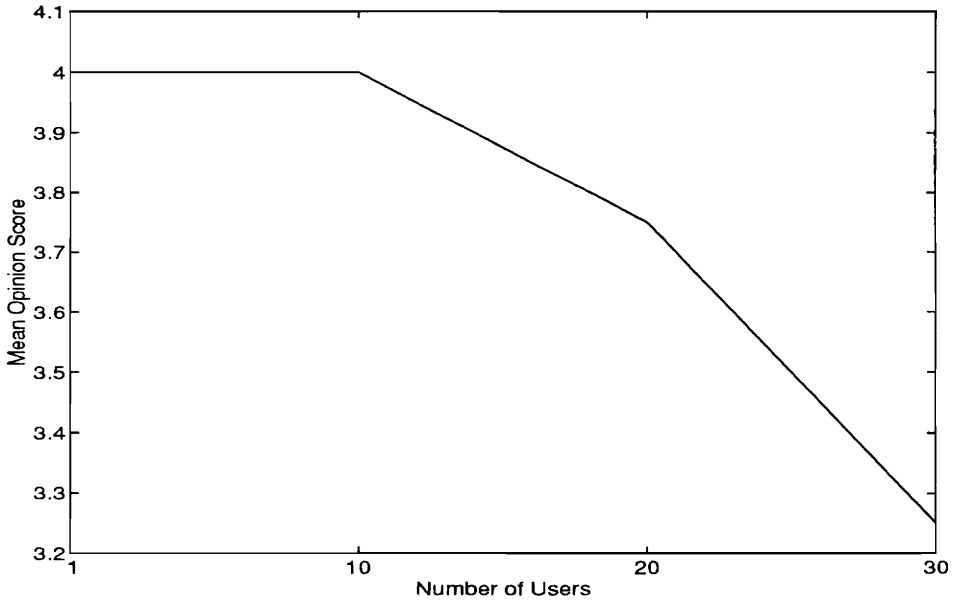

Figure 5.15: MOS - Male Speech, Forward Channel, Urban Environment, and 10dB channel SNR

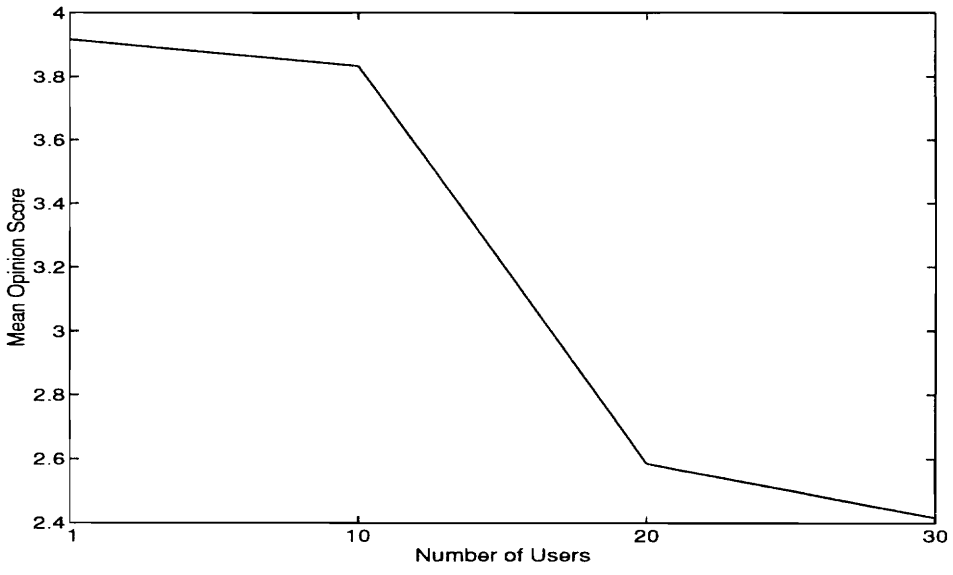

Figure 5.16: MOS - Male Speech, Reverse Channel, Urban Environment, and 10dB channel SNR

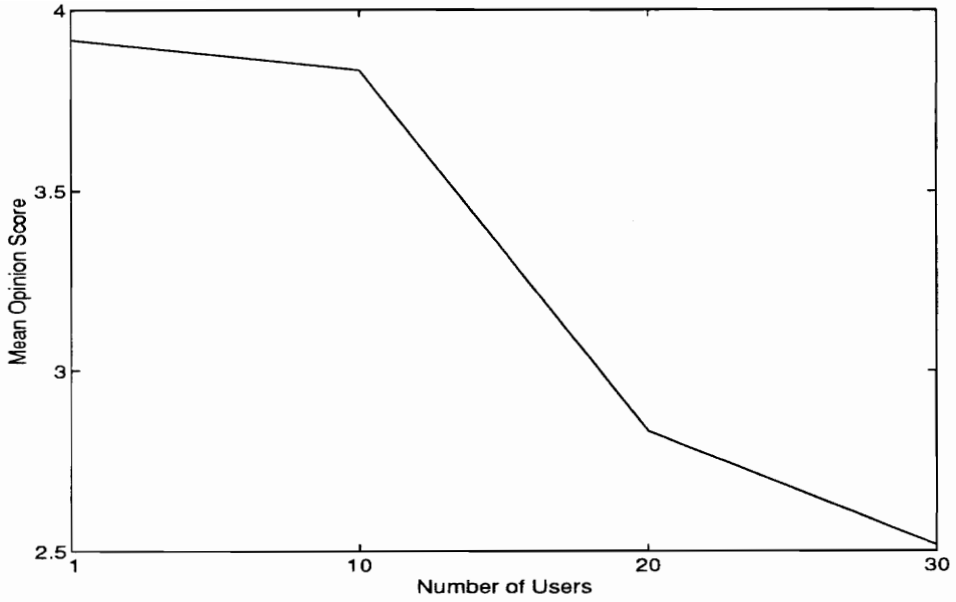

Figure 5.17: MOS - Male Speech, Forward Channel, Urban Environment, and 5dB channel SNR

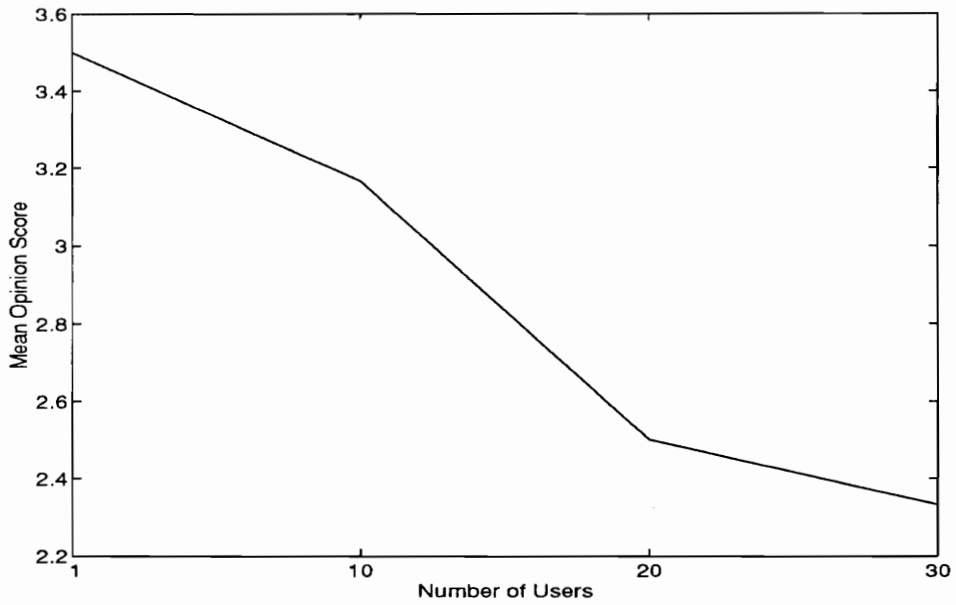

Figure 5.18: MOS - Male Speech, Reverse Channel, Urban Environment, and 5dB channel SNR

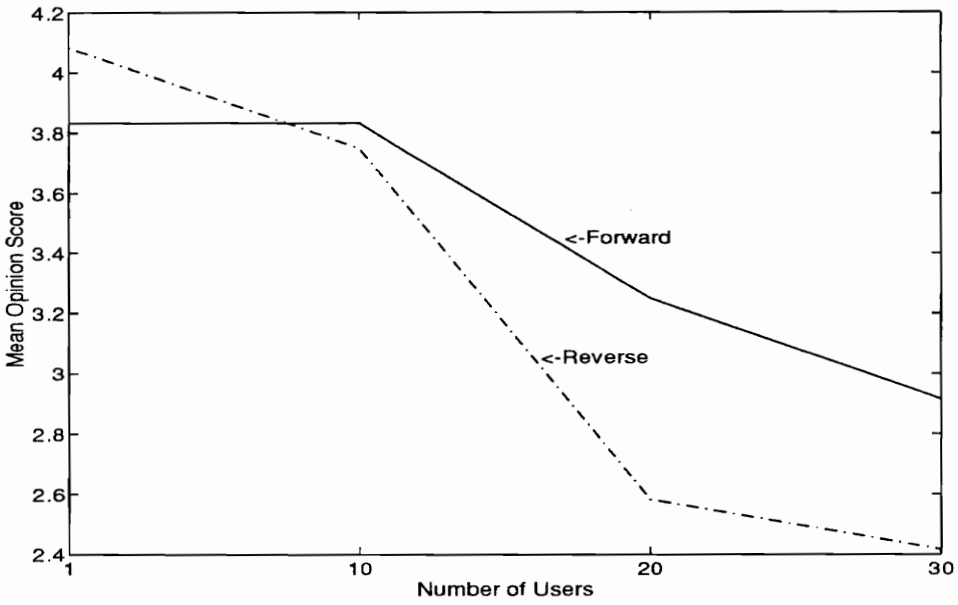

Figure 5.19: MOS comparison - Male Speech, Rural Environment, and 10dB channel SNR

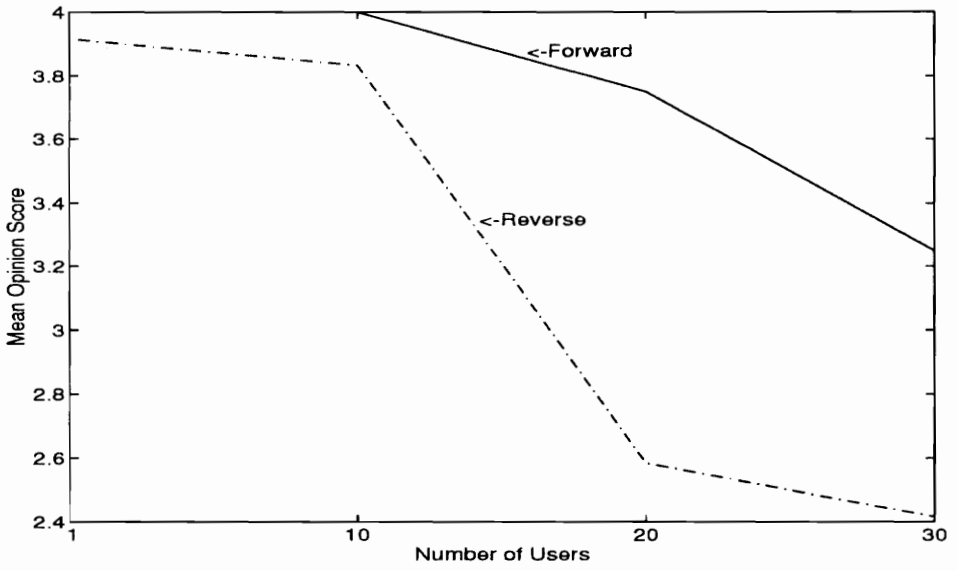

Figure 5.20: MOS comparison - Male Speech, Urban Environment, and 10dB channel SNR

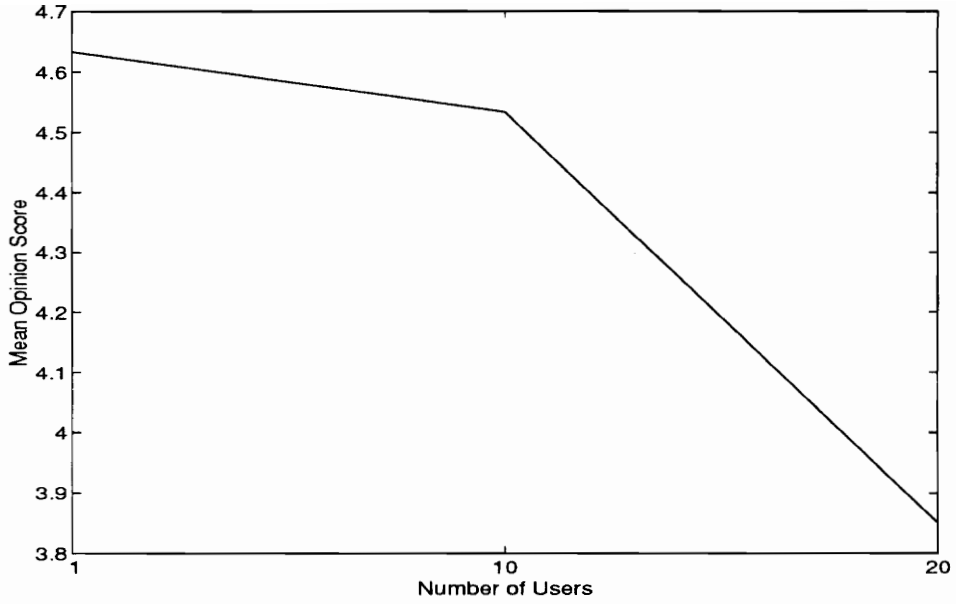

Figure 5.21: MOS - Male Speech, Forward Channel, 1 Ray Rayleigh Fading

Figures 5.21 and 5.22 show the MOS for male speech outputs for 1 Ray Rayleigh Fading. The speech quality for 1 Ray Rayleigh Fading is observed to be close to that for rural environment, although it is slightly worse.

Figures 5.23 and 5.24 show the MOS for male speech outputs for 2 Ray Rayleigh Fading. Performance in 2 Ray Rayleigh Fading is worse compared to 1 Ray Rayleigh Fading. Nevertheless, the speech quality is still acceptable for reasonable number of users.

5.3 Objective Measures

The objective measures used to assess the performance of the vocoder were the signal-to-noise ratio (SNR) and the Segmental SNR (SSNR). The former is calculated over the whole length of speech whereas the latter is calculated frame by frame.

The SNR is determined as follows.

$$SNR = 10 \log_{10} \frac{\sum_{i=1}^{n} s_{in_i}^2}{\sum_{i=1}^{n} (s_{out_i} - s_{in_i})^2} \tag{5.1}$$

where n is the number of samples and s_{in} and s_{out} are the input speech.

The Segmental SNR is determined by calculating the SNR for every frame and averaging

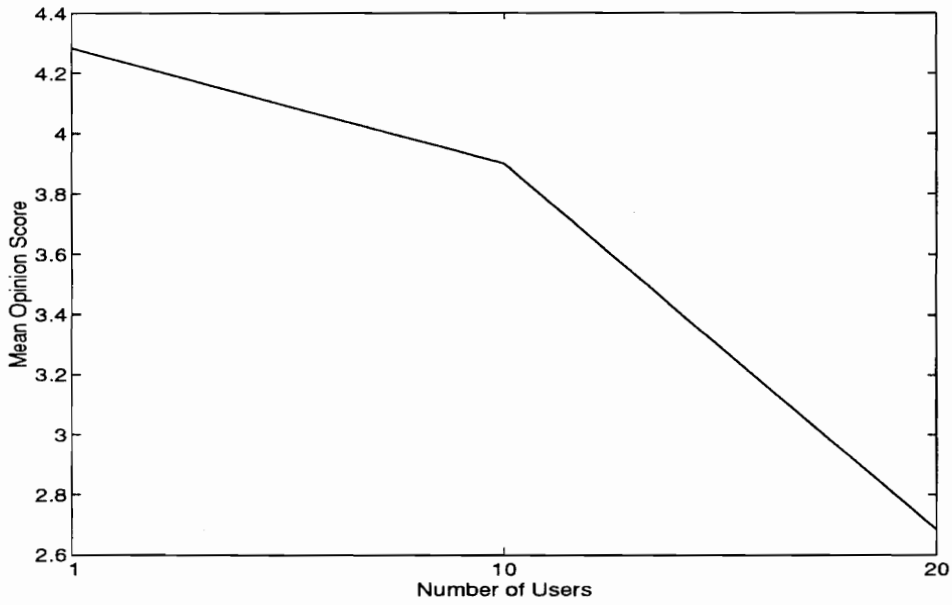

Figure 5.22: MOS - Male Speech, Reverse Channel, 1 Ray Rayleigh Fading

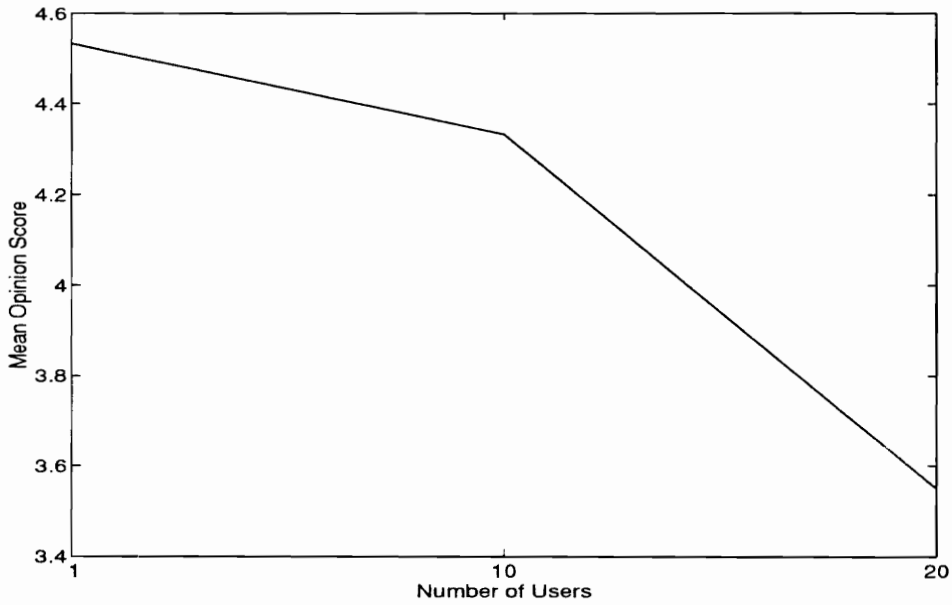

Figure 5.23: MOS - Male Speech, Forward Channel, 2 Ray Rayleigh Fading

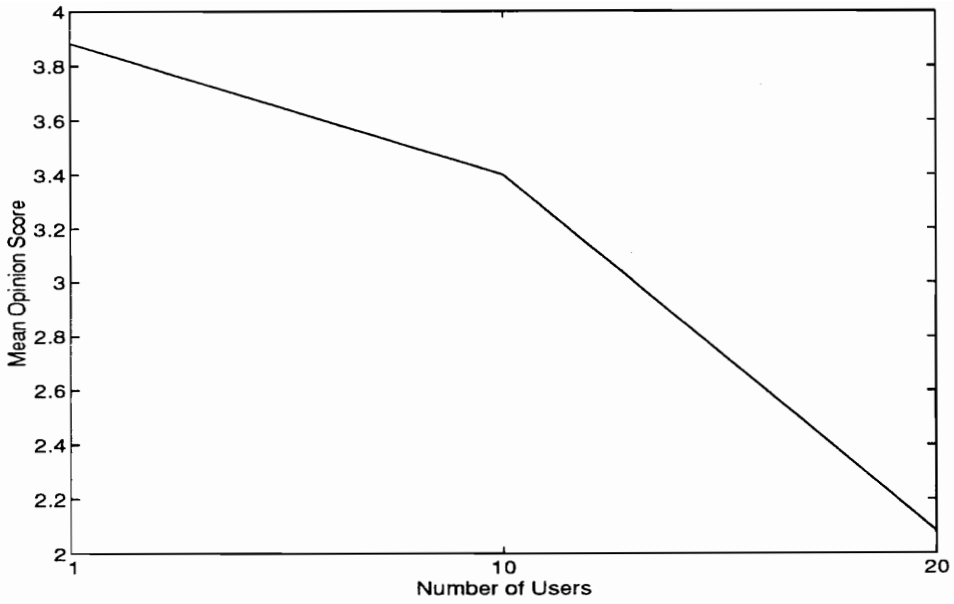

Figure 5.24: MOS - Male Speech, Reverse Channel, 2 Ray Rayleigh Fading

these values over the total number of frames in the input speech. The SSNR is computed as

$$SSNR = \frac{1}{F}\sum_{i=1}^{F} SNR_i \tag{5.2}$$

where F is the total number of frames and SNR_i is the signal-to-noise ratio of the i^{th} frame and is calculated as in Equation 5.1 with n being the number of samples in each frame.

The SNR calculated over the entire speech segment as in Equation 5.1 fails to paint an accurate picture of the quality of speech, as can be observed from the following Figures 5.25 and 5.26 which show the MOS results and SNR values for a particular speech output. This can be partly attributed to the fact that the SNR does not take into account speech frames which have been reconstructed perfectly, i. e. , with no error. Also the SNR weights the sections of speech having larger power (like vowels) more than the smaller-power sections (like consonants, transient periods, etc.), which inspite of their significance cannot reflect the SNR as well as a larger-power section. Hence, the SNR as given by Equation 5.1 was not considered to be a good speech quality evaluation tool.

Figures 5.27 and 5.28 show the Segmental SNR for female speech outputs in a rural environment at 10dB channel SNR. These scores tally well with the corresponding MOS

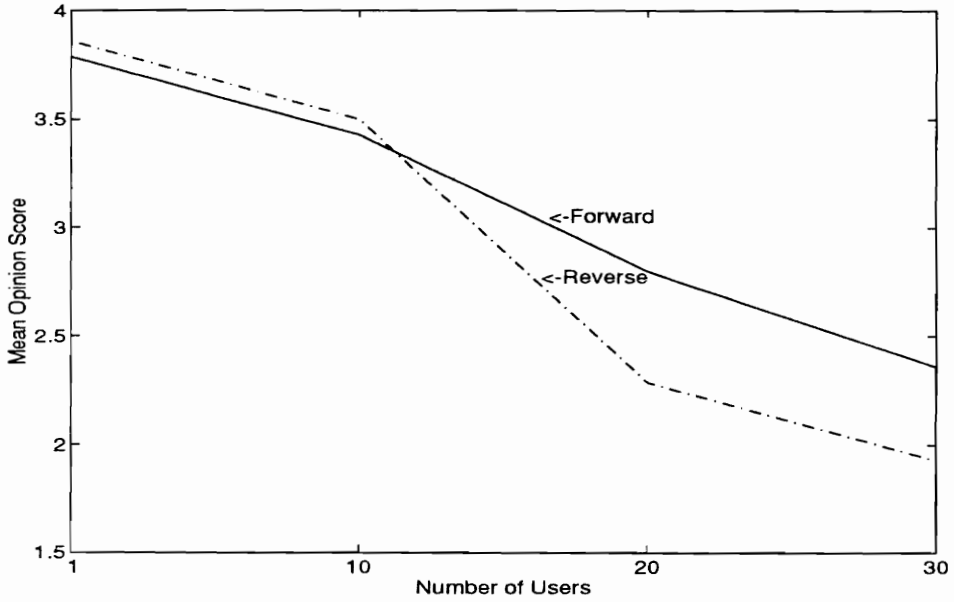

Figure 5.25: MOS results for 5dB channel SNR, Reverse Channel

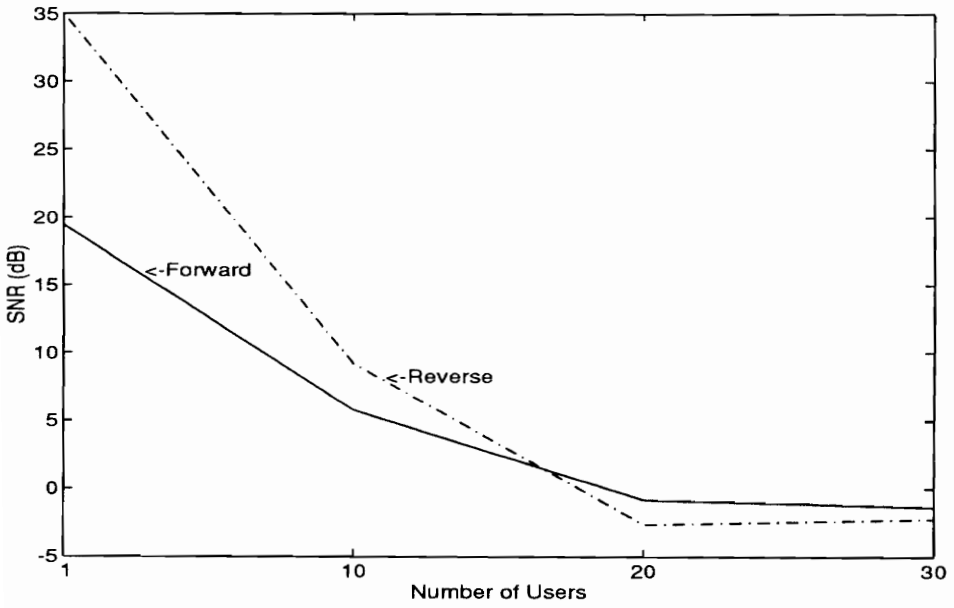

Figure 5.26: Classic SNR values for 5dB channel SNR, Reverse Channel

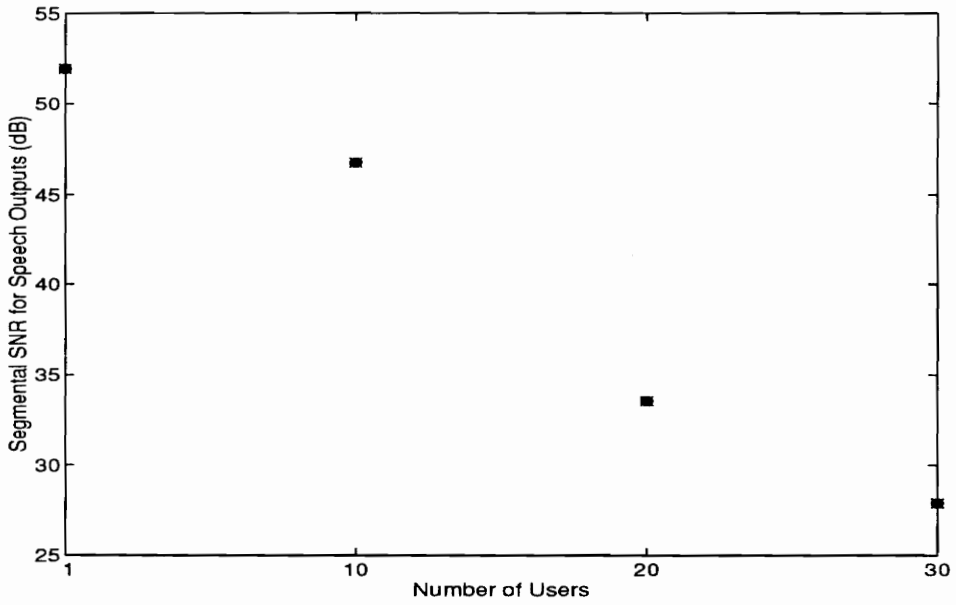

Figure 5.27: SSNR - Female Speech, Forward Channel, Rural Environment, and 10dB channel SNR

values in the sense that they have the same trend as the MOS.

Figures 5.29 and 5.30 show the Segmental SNR for female speech outputs in an urban environment at 10dB channel SNR. The SSNR is worse compared to that in the rural environment. The SSNR decreases more rapidly as the number of users increases.

Figures 5.31 and 5.32 show the Segmental SNR for female speech outputs in an urban environment at 5dB channel SNR. As in the case of MOS measure, the speech quality degrades slightly as the channel SNR decreases. The speech has tolerable distortion since the SSNR values are reasonably high.

Figure 5.33 compares the Segmental SNR for female speech outputs for both the channels in a rural environment at 10dB channel SNR. Again, it is observed that the forward channel has higher SSNR for a larger number of users.

Figure 5.34 compares the Segmental SNR for female speech outputs for both the channels in an urban environment at 10dB channel SNR. Once more, the forward channel is observed to have better SSNR than the reverse channel for large numbers of users.

Figures 5.35 and 5.36 compare the Segmental SNR for female speech outputs for 1 Ray Rayleigh Fading, while Figures 5.37 and 5.38 show the SSNR for 2 Ray Rayleigh Fading.

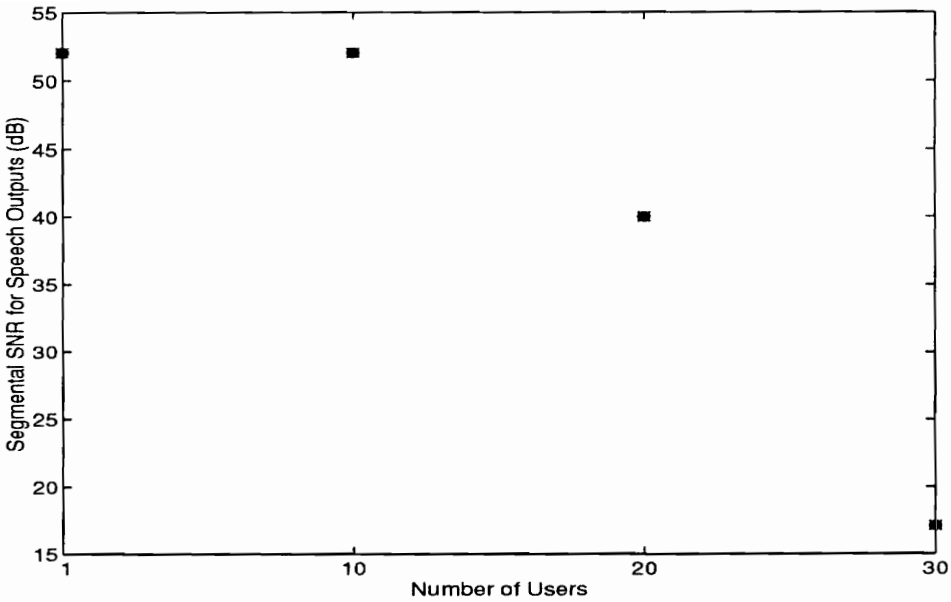

Figure 5.28: SSNR - Female Speech, Reverse Channel, Rural Environment, and 10dB channel SNR

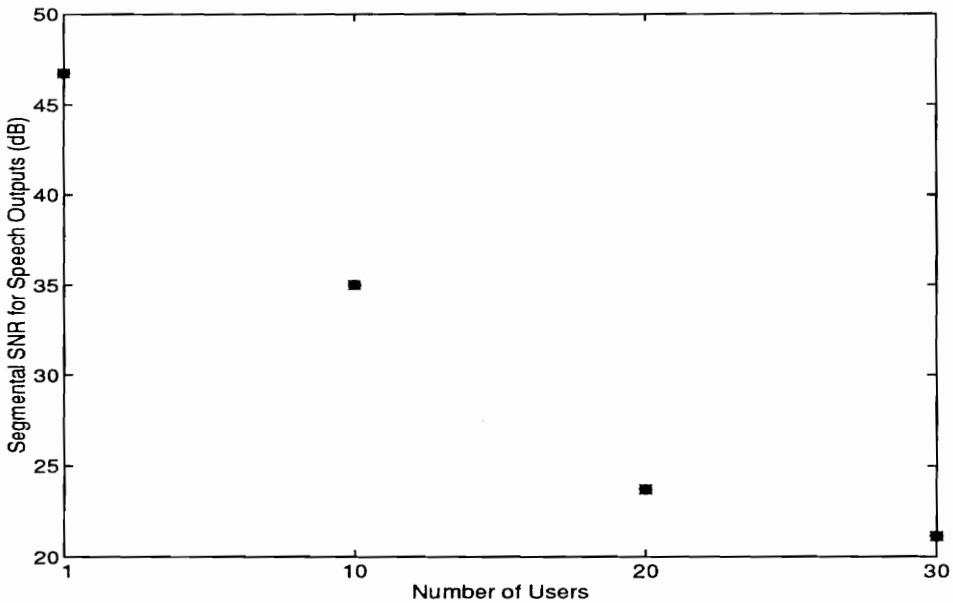

Figure 5.29: SSNR - Female Speech, Forward Channel, Urban Environment, and 10dB channel SNR

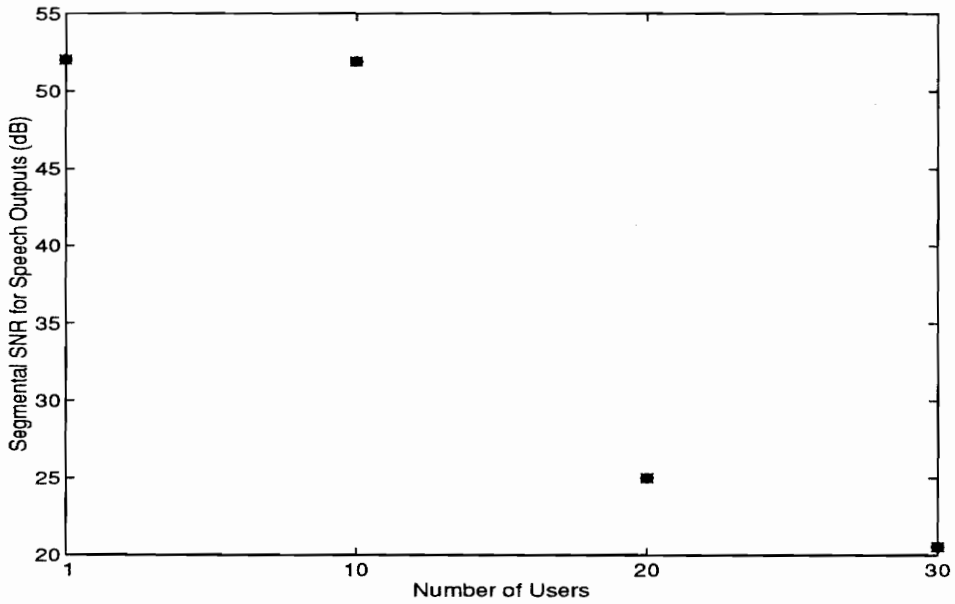

Figure 5.30: SSNR - Female Speech, Reverse Channel, Urban Environment, and 10dB channel SNR

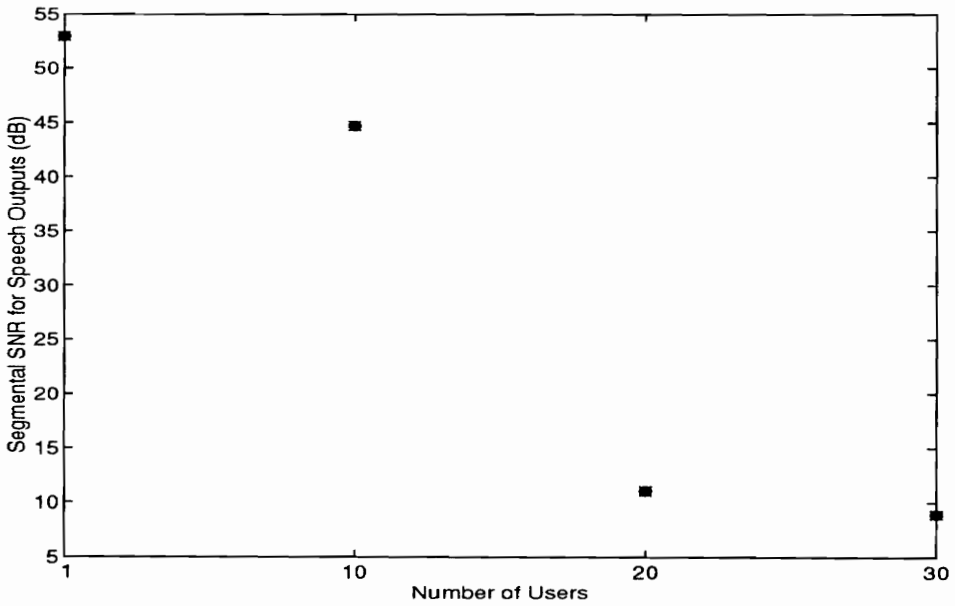

Figure 5.31: SSNR - Female Speech, Forward Channel, Urban Environment, and 5dB channel SNR

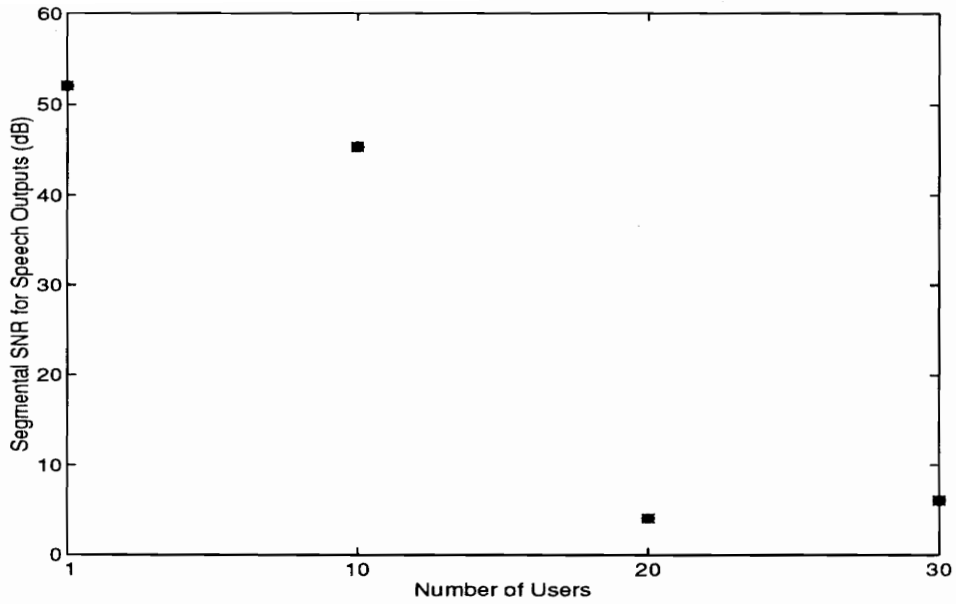

Figure 5.32: SSNR - Female Speech, Reverse Channel, Urban Environment, and 5dB channel SNR

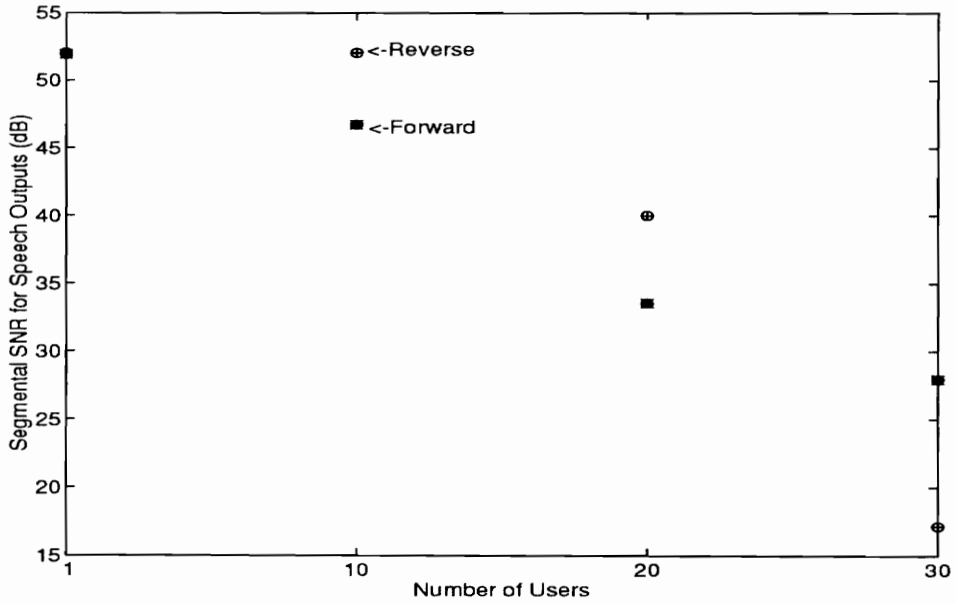

Figure 5.33: SSNR comparison - Female Speech, Rural Environment, and 10dB channel SNR

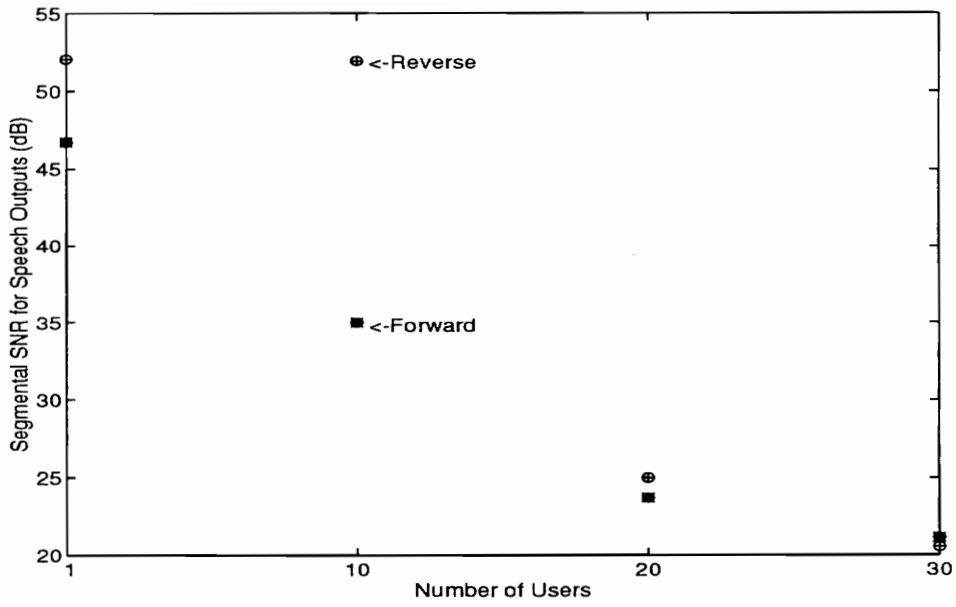

Figure 5.34: SSNR comparison - Female Speech, Urban Environment, and 10dB channel SNR

The SSNR for the 1 Ray case is seen to be higher than that for the 2 Ray case, as was observed in the MOS results. However, the SSNR is high enough to guarantee acceptable speech quality.

As can be seen these results tally well with the MOS results shown in the previous section. So the Segmental SNR is observed to be a good objective measure to estimate the perceptual quality of speech. In circumstances which do not allow subjective speech quality assessment, Segmental SNR could be considered to provide an intuitive idea into the perceptibility of the speech.

The Figures 5.39 to 5.50 show the corresponding values of SSNR for male speech. It is observed that the male speech has lower SSNR than the female speech, which is in agreement with the MOS results. The SSNR is seen to decrease as the number of users increases, which is the same trend as in the case of female speech.

Figures 5.39 and 5.40 show the Segmental SNR for male speech outputs in a rural environment at 10dB channel SNR and Figures 5.41 to 5.44 show the Segmental SNR in urban environments at 10dB and 5dB channel SNR. The SSNR for the rural environments are seen to be higher than that for the urban environments.

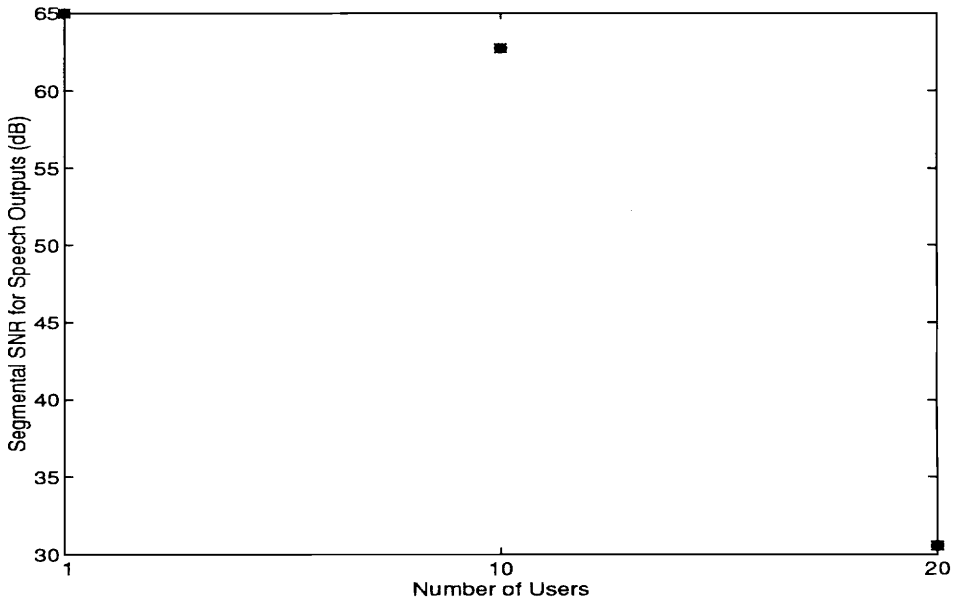

Figure 5.35: SSNR - Female Speech, Forward Channel, 1 Ray Rayleigh Fading

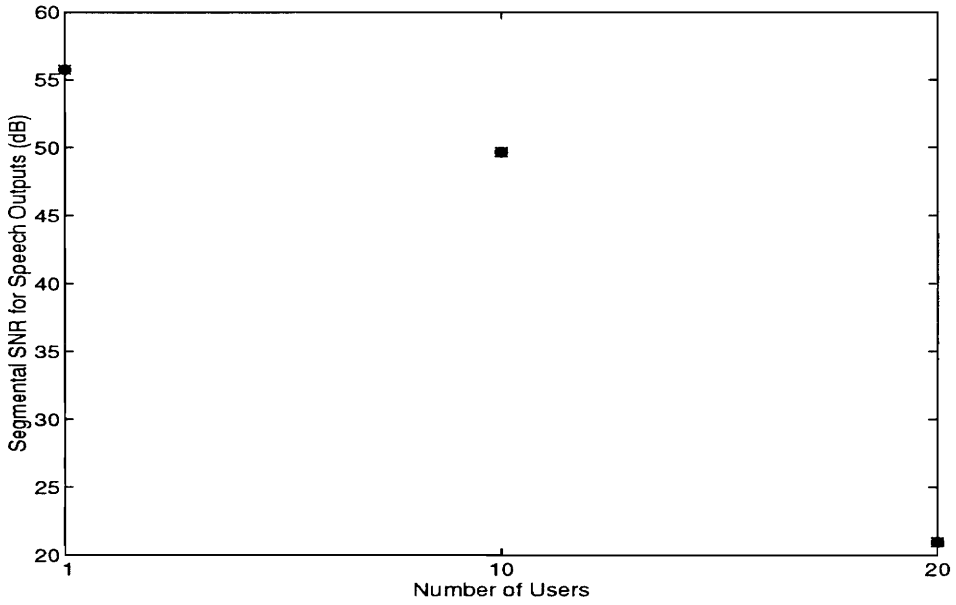

Figure 5.36: SSNR - Female Speech, Reverse Channel, 1 Ray Rayleigh Fading

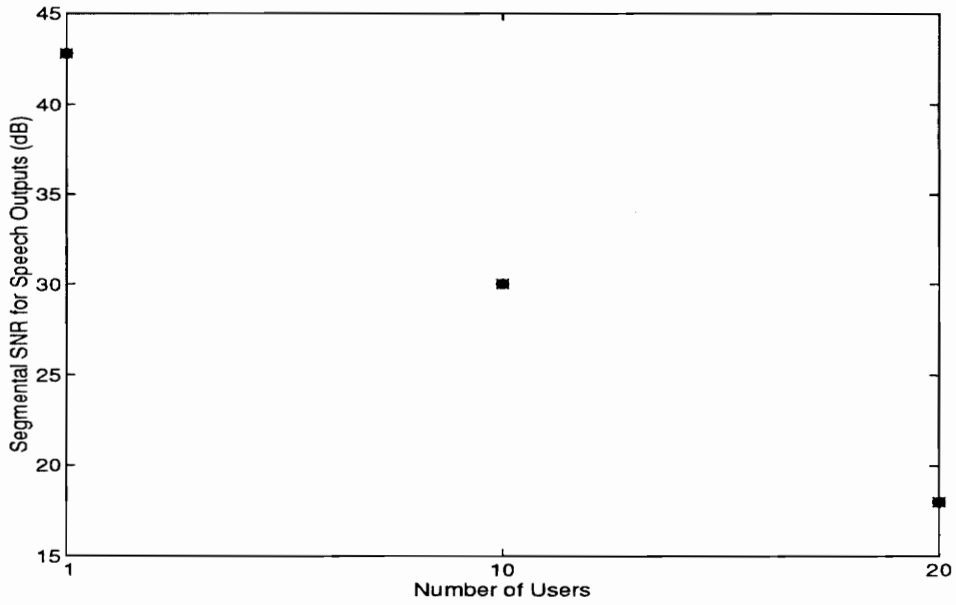

Figure 5.37: SSNR - Female Speech, Forward Channel, 2 Ray Rayleigh Fading

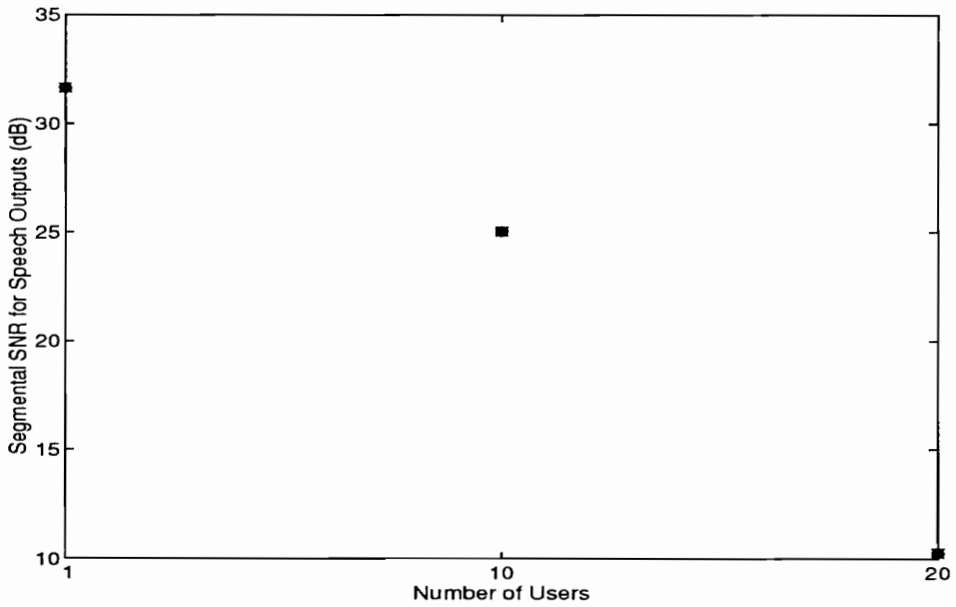

Figure 5.38: SSNR - Female Speech, Reverse Channel, 2 Ray Rayleigh Fading

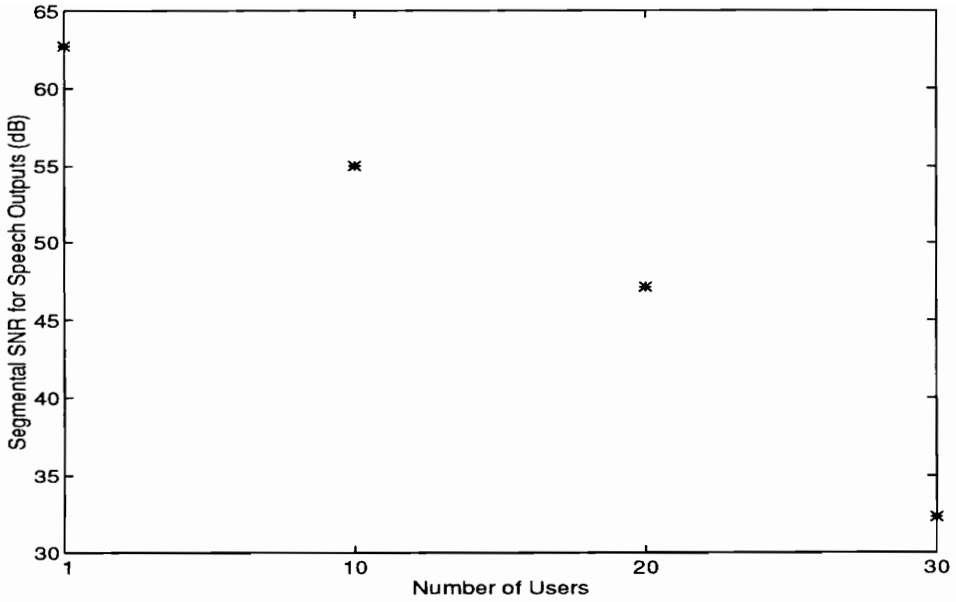

Figure 5.39: SSNR - Male Speech, Forward Channel, Rural Environment, and 10dB channel SNR

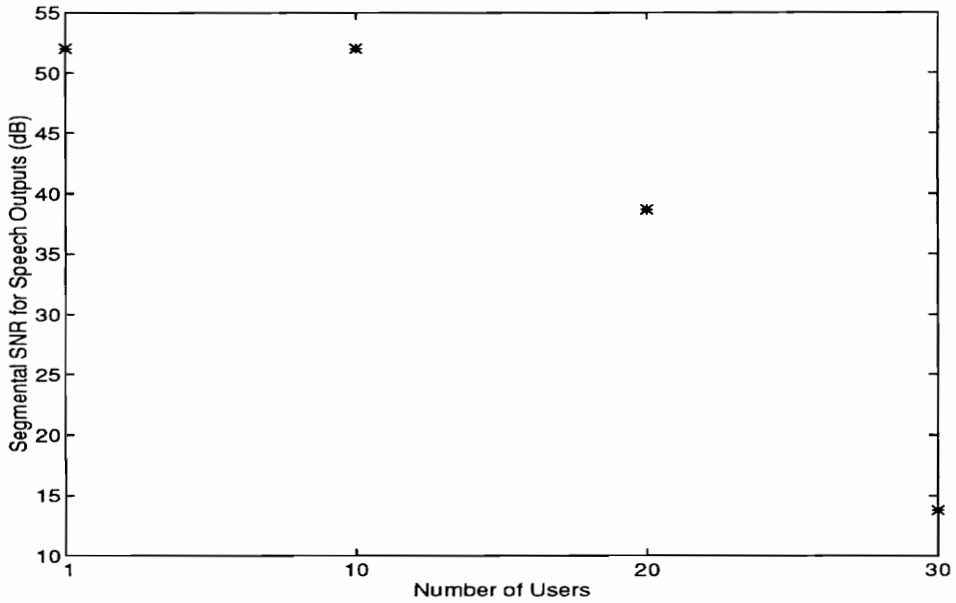

Figure 5.40: SSNR - Male Speech, Reverse Channel, Rural Environment, and 10dB channel SNR

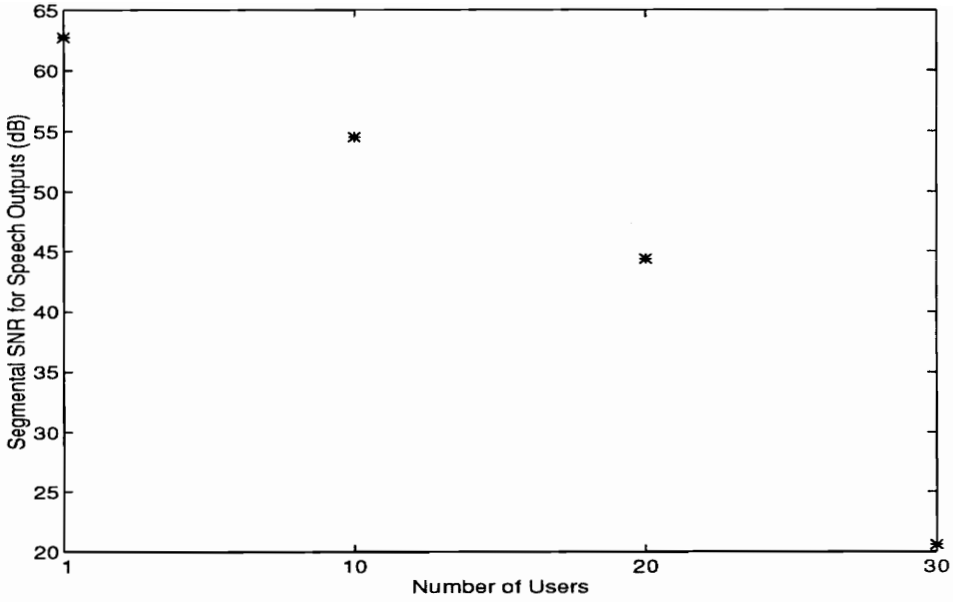

Figure 5.41: SSNR - Male Speech, Forward Channel, Urban Environment, and 10dB channel SNR

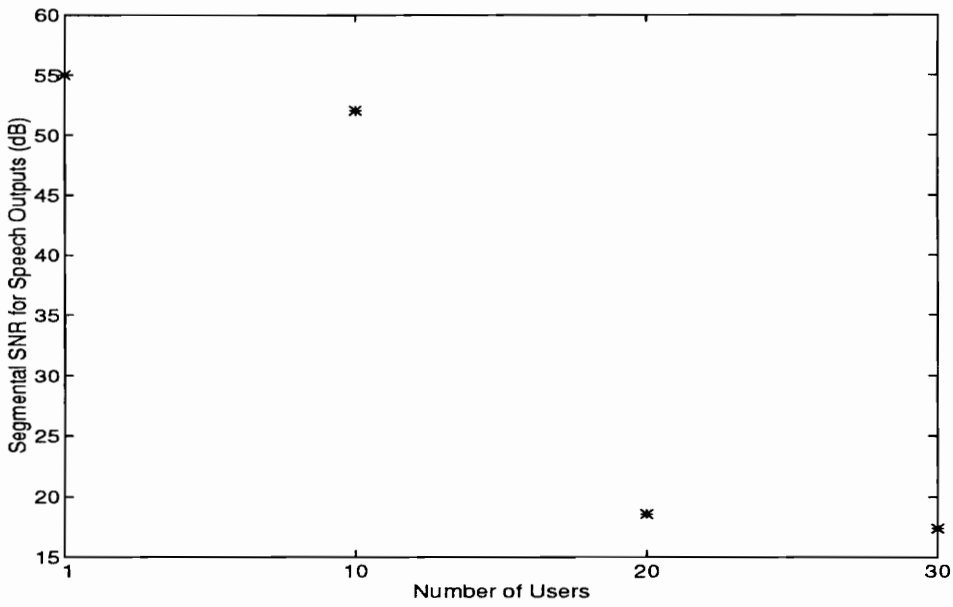

Figure 5.42: SSNR - Male Speech, Reverse Channel, Urban Environment, and 10dB channel SNR

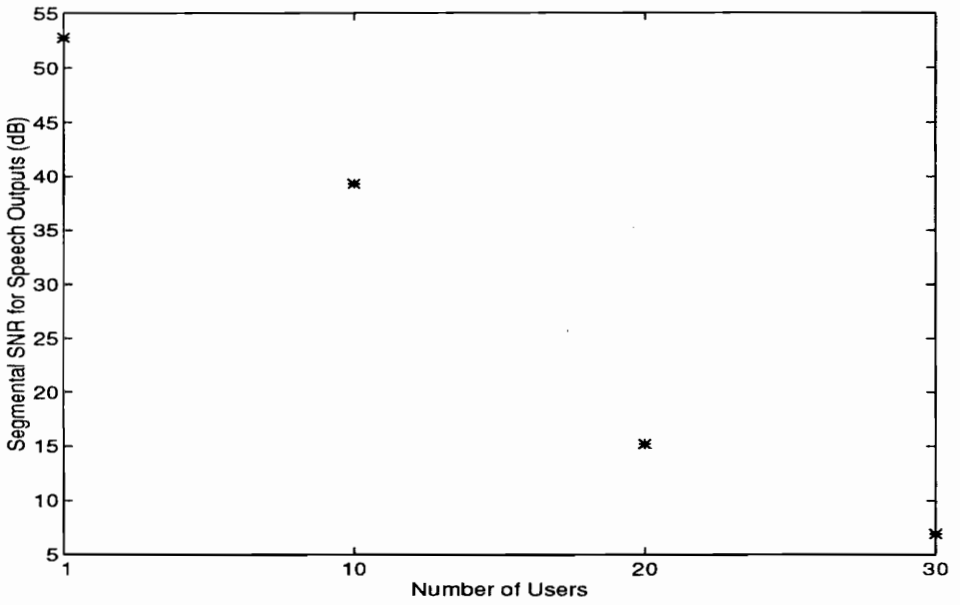

Figure 5.43: SSNR - Male Speech, Forward Channel, Urban Environment, and 5dB channel SNR

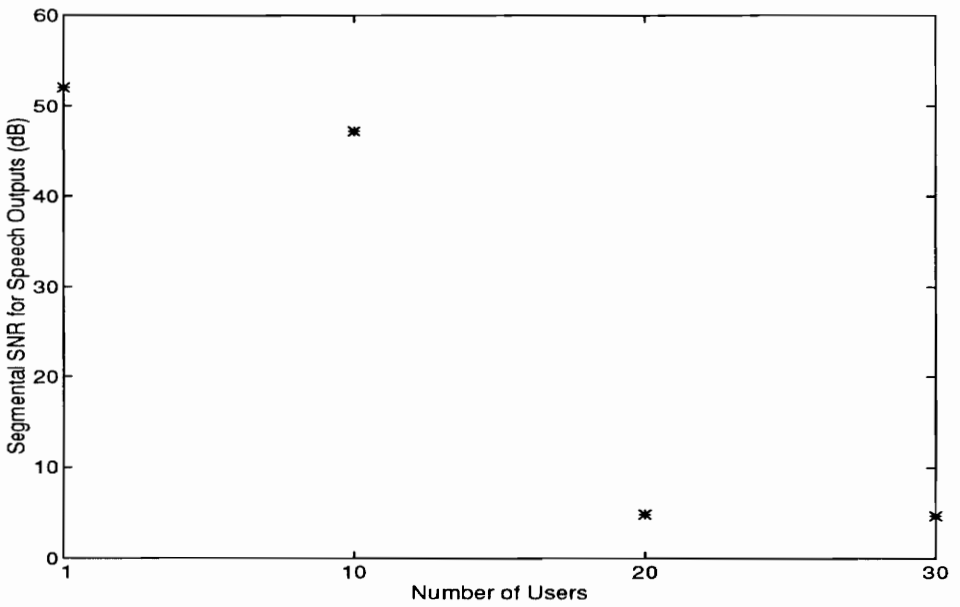

Figure 5.44: SSNR - Male Speech, Reverse Channel, Urban Environment, and 5dB channel SNR

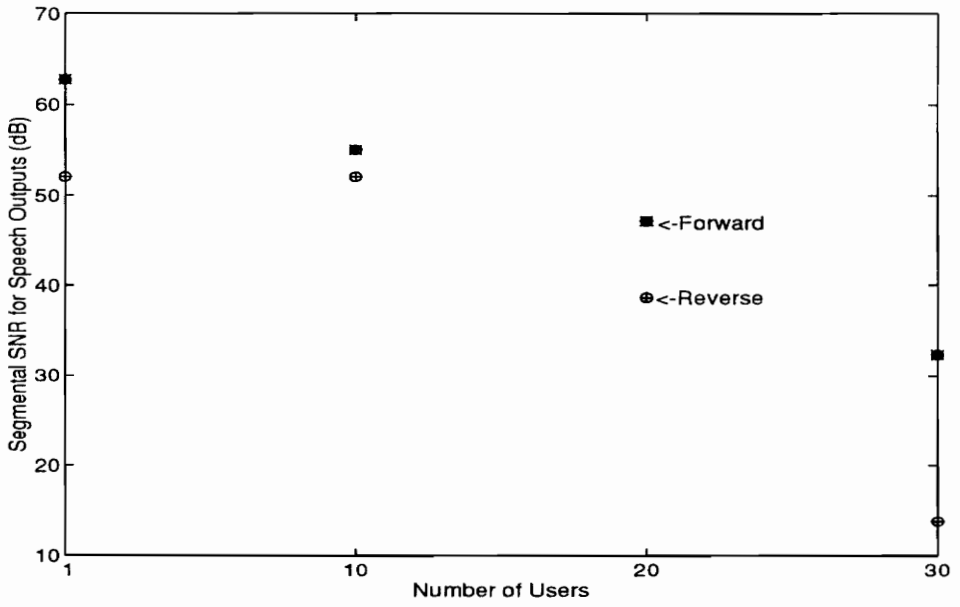

Figure 5.45: SSNR comparison - Male Speech, Rural Environment, and 10dB channel SNR

Figures 5.45 and 5.46 compare the Segmental SNR for male speech outputs for both the channels in a rural environment and an urban environment respectively, at 10dB channel SNR. The forward channel is again seen to have a higher SSNR for large numbers of users.

Figures 5.47 and 5.48 compare the Segmental SNR for male speech outputs for 1 Ray Rayleigh Fading, while Figures 5.49 and 5.50 show the same for 2 Ray Rayleigh Fading. It is observed, as in the case of female speech that, the output for 1 Ray Fading has higher SSNR than the output for 2 Ray Fading.

Lastly, the Coherence Function (CF) is briefly discussed as another useful objective measure [30]. The Coherence Function, which is a frequency domain measure, is evaluated as

$$cf = 10 \log_{10} \frac{\gamma(f)^2}{1 - \gamma(f)^2} \tag{5.3}$$

where the numerator is the coherent output spectrum and the denominator is the noncoherent output spectrum.

$\gamma(f)^2$ is evaluated as

$$\gamma(f)^2 = \frac{|\sum X(f)Y^*(f)|^2}{\sum |X(f)|^2 \sum |Y(f)|^2}$$

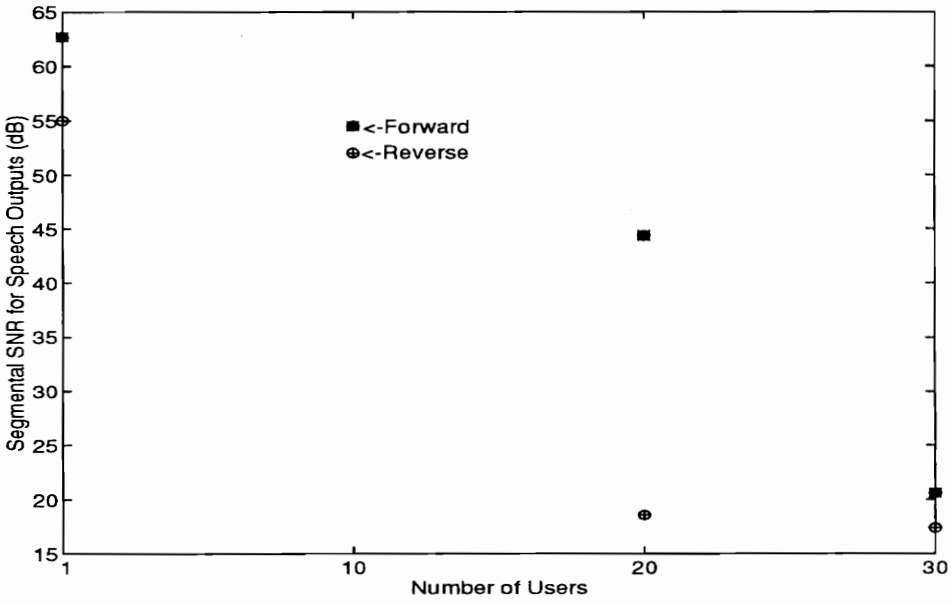

Figure 5.46: SSNR comparison - Male Speech, Urban Environment, and 10dB channel SNR

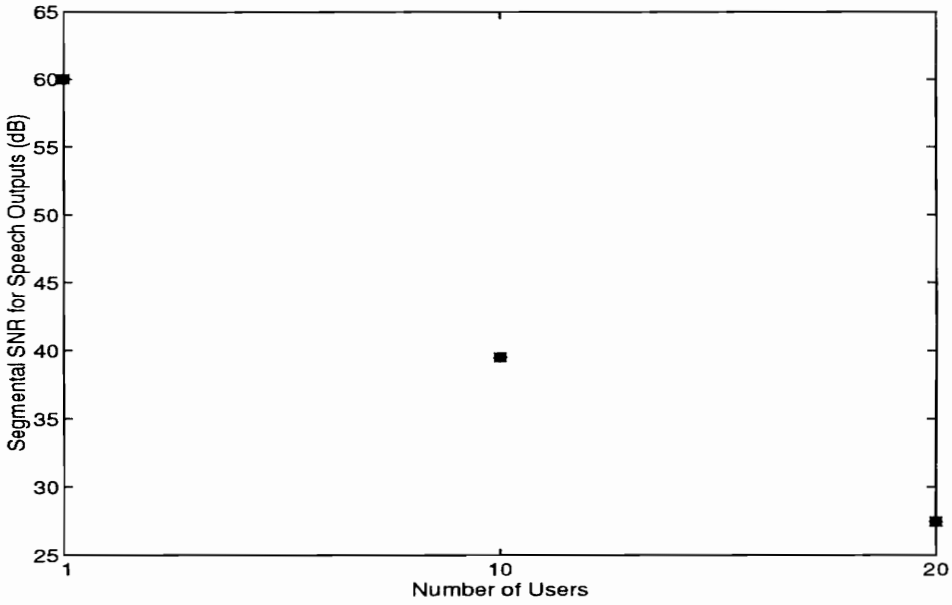

Figure 5.47: SSNR - Male Speech, Forward Channel, 1 Ray Rayleigh Fading

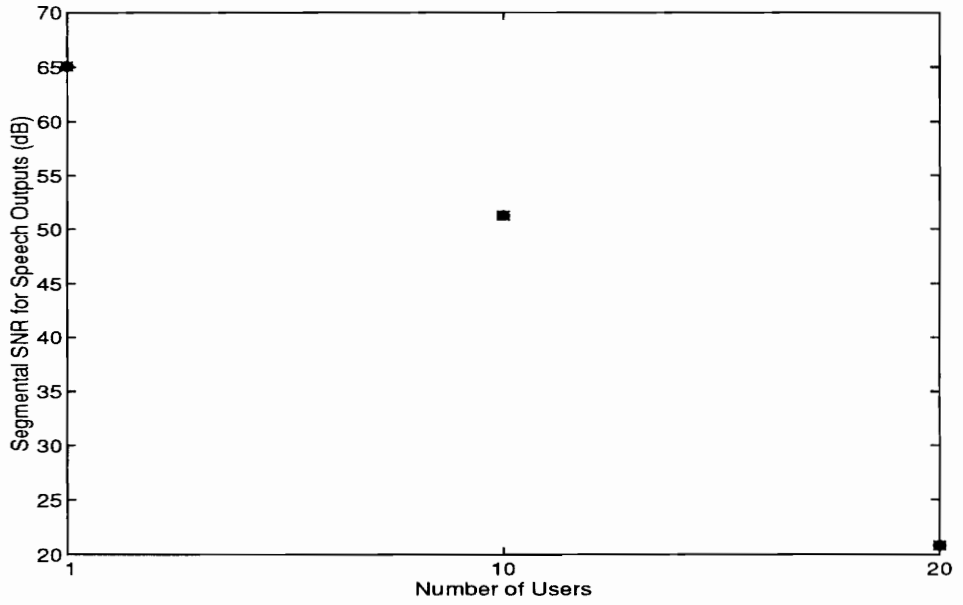

Figure 5.48: SSNR - Male Speech, Reverse Channel, 1 Ray Rayleigh Fading

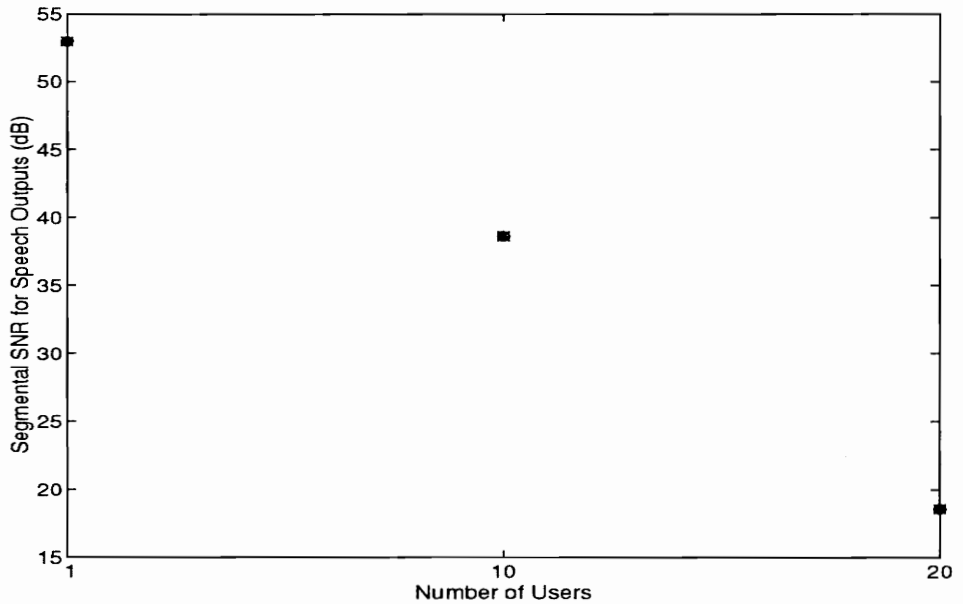

Figure 5.49: SSNR - Male Speech, Forward Channel, 2 Ray Rayleigh Fading

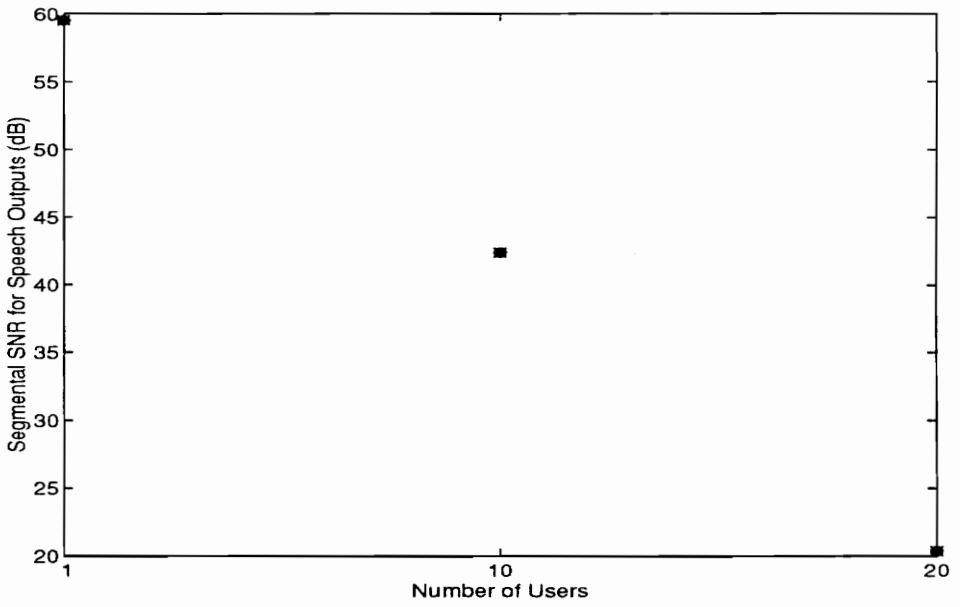

Figure 5.50: SSNR - Male Speech, Reverse Channel, 2 Ray Rayleigh Fading

Table 5.2: CF for Female Speech - Forward Channel

Users	Rural	Urban	1 Ray Rayleigh Fading	2 Ray Rayleigh Fading
1	15.04	15.04	16.80	15.00
10	14.61	12.79	15.00	09.37
20	14.37	02.37	03.27	00.00
30	02.00	00.32	-	-

where $X(f)$ and $Y(f)$ are obtained by applying the Fast-Fourier Transform to the input and output signal respectively.

The CF descibes the correlation between the input speech spectrum and the output speech spectrum. High values of CF indicate good correlation between the input and output spectrums and hence indicate how close the output speech is to the input speech.

The values of the Coherence Function are listed in Tables 5.2, 5.3, 5.4 and 5.5 for the forward and reverse CDMA channels for both female and male speeches at 10dB channel SNR. These values agree well with the SSNR values. The trends observed are the same.

It can be seen from these values that the performance falls as the numbers of users increase. The CF for the rural environment is higher than that for the urban environment as

Table 5.3: CF for Female Speech - Reverse Channel

Users	Rural	Urban	1 Ray Rayleigh Fading	2 Ray Rayleigh Fading
1	15.00	19.56	15.00	03.28
10	15.00	04.88	09.26	02.28
20	04.12	00.00	00.00	00.46
30	02.00	00.00	-	-

Table 5.4: CF for Male Speech - Forward Channel

Users	Rural	Urban	1 Ray Rayleigh Fading	2 Ray Rayleigh Fading
1	24.34	24.34	21.75	12.49
10	16.96	15.46	15.00	12.12
20	13.33	08.44	05.19	01.57
30	02.66	00.30	-	-

Table 5.5: CF for Male Speech - Reverse Channel

Users	Rural	Urban	1 Ray Rayleigh Fading	2 Ray Rayleigh Fading
1	22.38	15.00	37.74	16.82
10	15.40	15.00	13.56	9.35
20	04.93	10.76	03.96	01.15
30	00.00	02.32	-	-

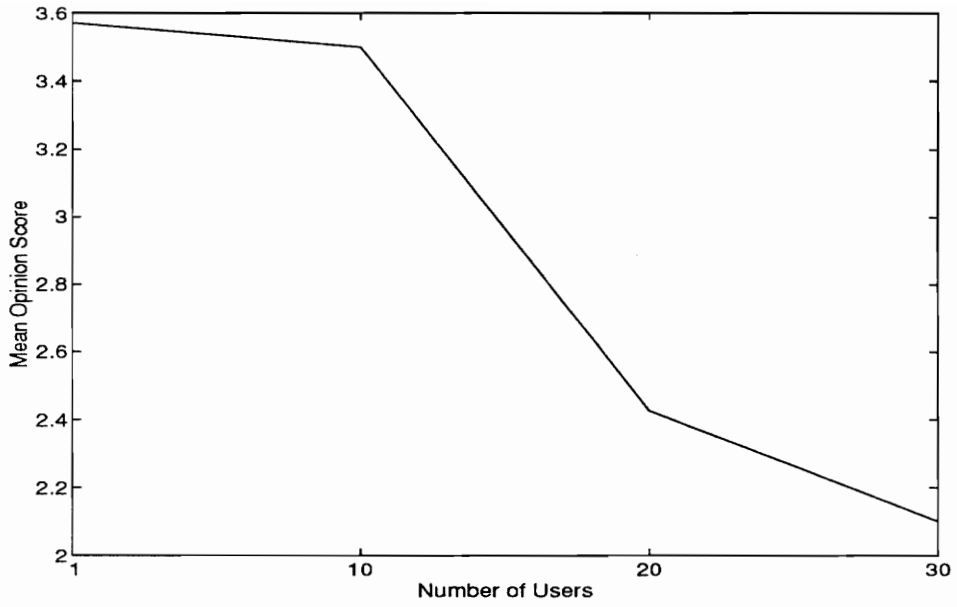

Figure 5.51: MOS for Female Output Speech in an Urban Environment, Forward Channel

was the trend with the MOS values. The performance in 1 Ray Rayleigh fading environment is close to that in the rural environment whereas the performance in 2 Ray Rayleigh fading is worse.

Figures 5.51 and 5.52 show the MOS values and CF values respectively for the female speech in an urban environment. The trends indicated by both of these curves are similar. It could hence be concluded that the CF values also track the subjective quality assessment results closely.

5.4 Summary

This chapter summarized the simulation specifics and the results for the performance of the QCELP vocoder under different channel conditions.

It was observed that the female speech was reproduced with better quality than the male speech. This could probably be attributed partly to the female speech being clearer and spoken slower than the male speech.

It was observed that the speech was reproduced with only very faint distortion for bit error rates below 0.009%, without passing through the CDMA system. With the CDMA

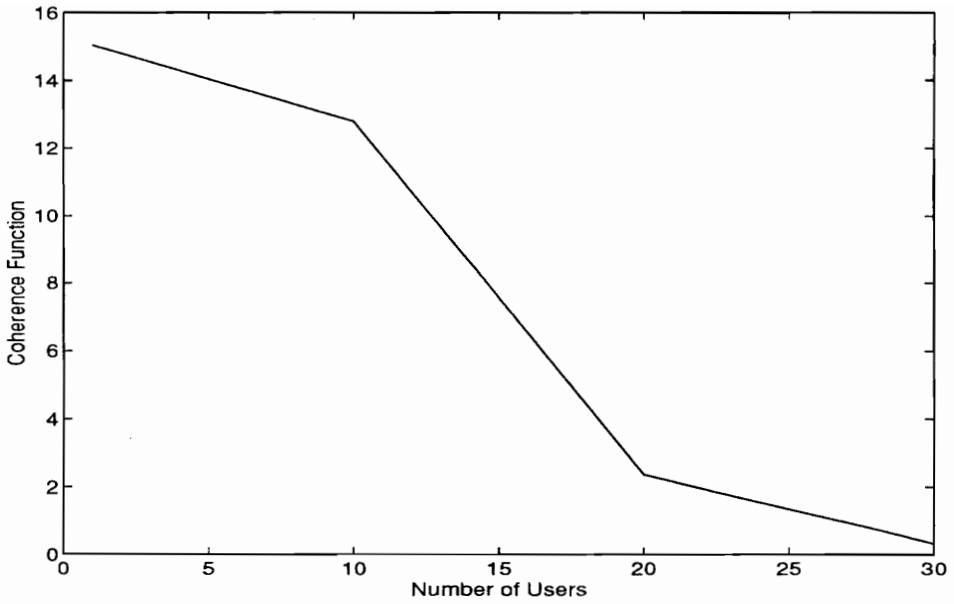

Figure 5.52: CF for Female Output Speech in an Urban Environment, Forward Channel

system present, because of the various spreading steps and the convolutional coding, the vocoder could tolerate far higher channel bit error rates.

It was also observed that the interleaving in the CDMA system helped reduce the effect of bursty errors caused by fading to a large extent and hence the effect of these channel errors were not observed to significantly distort the speech. The speech from the Rayleigh Fading environments were as good as the speech from the rural environments. The forward channel performed better than the reverse channel under heavy loading (large number of users). The reverse channel was superior to the forward channel when there were less number of users.

The objective quality measures, SSNR and CF, produced results which were in close agreement with the MOS results.

Chapter 6

Conclusions

6.1 Summary of the Research

A simulation of Qualcomm's variable rate CELP vocoder for use with the CDMA system specified in the IS-95 standard has been presented in this thesis. The variable rate vocoder was implemented in software and its performance was analyzed for various channel conditions.

The average output bit rate from the vocoder for the speech inputs was observed to be in the range 5.7 kbps to 7.2 kbps. This could probably reduce further in actual telephone communications because of the low voice activity factor.

As was observed in Chapter 4, the speech outputs corresponding to different channels in the CDMA system are distorted in the sense that the time waveforms of the outputs are different from the input speech waveform, but not to a great extent. The amount of distortion observed was close to what was expected.

Rayleigh fading caused burst errors in the packet data but owing to the interleaving action in the CDMA transmitter and de-interleaving in the receiver, the effect of this fading is almost eliminated. The spreading process which increases the transmitted signal bandwidth, also helped in reducing the effects of fading. Even if there remains a trace of some burst errors, this error would be either spread over only one or two packets. Since each packet is of only 20 ms, the effect of the fading converts to a barely perceptible distortion.

The speech outputs for the urban and rural environments whose channel impulse responses were measured by Bell Atlantic, were also obtained, for different number of users. The CDMA simulation program written by Yingjie Li was used to obtain the corresponding bit-by-bit error patterns with which the packet data were XORed.

The speech degradation is negligible upto 20 users, though a light degradation was observed for 30 users and above. The speech was still intelligible with 30 users; however,

some may find the amount of distortion objectionable.

The speech quality for the forward channel, on average, was observed to be consistently better than for the reverse channel, which was also expected because of the orthogonal spreading of different users inputs. For very small number of users the reverse channel performed better due to the heavy error correction coding used.

Channel SNRs of 10dB and above allowed excellent speech quality, and 5dB SNR allowed reasonably good speech quality too.

The packets of Rate 1 and 1/2 were observed to contribute more and more to speech quality and any significant number of errors in these packets caused the speech to distort much more than if the errors were caused in the lower rate packets.

Performance analysis of the vocoder was based on estimates of perceptual speech quality. Both subjective and objective measures were used to determine the quality. Mean Opinion Score was used as the subjective measure with as many as 12 listeners contributing their opinions on the quality of speech. Segmental SNR was established to have very good correspondence to the MOS results. The classic SNR was observed to provide a poor estimate of the actual perceptual speech quality. The Coherence Function was also calculated to obtain a spectral measure for the speech quality and these values corresponded well with the MOS results.

6.2 Future Work

The vocoder was simulated in isolation and the effects of the channels were incorporated offline using the bit-by-bit error patterns. A possible modification to this simulation would be to combine the vocoder and channel simulations, i. e. , to send each of the packets output by the encoder instantly to the channel after suitably coding, interleaving and spreading, and the demodulated and decoded received packet data bits into the decoder part of the vocoder to reconstruct the speech. This way the entire communication system would be available as a block for other simulations including varying the different parameters involved.

Different receiver structures could be used instead of the prescribed kind in the IS-95 standard. Either the plain RAKE receiver with different ways of calculating the decision statistic, or the suboptimal multistage detector which reduces the interference, could be used. The multistage RAKE receiver which would be a combination of the two, and is

currently under development in the MPRG, may greatly improve the speech quality.

Other objective measures like Itakura distance measure, Cepstral measures, Log Area Ratio measure, Log Likelihood measures, Weighted Spectral measures, etc. could be used to assess the quality of speech and compared with the MOS results. Subjective measures other than Mean Opinion Score such as rhyme tests and intelligibility tests could also be used.

The vocoder could also be implemented in hardware, using a DSP chip. This could pave the way for real time processing.

Bibliography

[1] R. Paget, *Human Speech: Some Observations, Experiments, and Conclusions as to the Nature, Origin, Purpose and Possible Improvement of Human Speech.* New York: Harcourt, 1930.

[2] A. N. Ince, "Overview of voice communications and speech processing," in *Digital Speech Processing* (A. N. Ince, ed.), Boston, Massachusetts: Kluwer Academic Publishers, 1992.

[3] H. E. Taub and D. L. Schilling, *Principles of Communication Systems.* McGraw-Hill, 1980.

[4] A. Gersho and R. M. Gray, *Vector Quantization and Signal Compression.* Kluwer Academic Publishers, 1992.

[5] B. P. Lathi, *Modern Digital and Analog Communication Systems.* The Dryden Press, 1989.

[6] W. C. Y. Lee, *Mobile Cellular Telecommunications.* McGraw-Hill, 1989.

[7] E. I. Association, "IS-54 cellular system dual-mode mobile station-base station compatibility standard," tech. rep., EIA/TIA, May 1990.

[8] Committee on Digital CDMA Cellular, "IS-95 wideband spread spectrum digital cellular system dual-mode mobile station — base station compatibility standard," tech. rep., EIA/TIA, TR45.5, Apr. 1992.

[9] N. S. Jayant, "Digital coding of speech waveforms: PCM, DPCM, and DM quantizers," *Proceedings of the IEEE*, vol. 62, pp. 611–632, May 1974.

[10] H. Dudley, "The vocoder," *Bell Labs Record*, vol. 17, pp. 122–126, 1939.

[11] R. H. Clarke, "A statistical theory of mobile radio reception," *Bell System Technical Journal*, pp. 957–1000, July 1968.

[12] L. W. Couch, *Digital and Analog Communication Systems.* New York: Macmillan Publishing Company, fourth ed., 1993.

[13] N. S. Jayant and P. S. Noll, *Digital Coding of Waveforms.* Englewood Cliffs, New Jersey: Prentice-Hall, 1984.

[14] S. P. Lloyd, "Least squares quantization in PCM," *IEEE Transactions on Information Theory*, vol. 28, pp. 129–137, Mar. 1982.

[15] J. G. Proakis, *Digital Communications.* New York: McGraw-Hill, second ed., 1989.

[16] V. K. Jaand R. E. Crochiere, "Quadrature mirror filter design in the time domain," *IEEE ICASSP*, vol. 32, pp. 353–361, Apr. 1984.

[17] J. R. Deller, J. G. Proakis, and J. H. Hansen, *Discrete-Time Processing of Speech Signals.* New York: Macmillan Publishing Company, first ed., 1993.

[18] B. P. Bogert, M. J. R. Healy, and J. W. Tukey, "The quefrency analysis of time series and echoes: Cepstrum, pseudo- autocovariance, cross-cepstrum and saphe cracking," *Proceeding of the Symposium on Time Series Analysis* (M. Rosenblatt, ed.), (New York), pp. 209–243, John Wiley & Sons, 1963.

[19] A. V. Oppenheim, "A speech analysis-synthesis system based on homomorphic filtering," *Journal of the Acoustical Society of America*, vol. 45, pp. 458–465, Feb. 1969.

[20] L. R. Rabiner and R. W. Schafer, *Digital Processing of Speech Signals.* Englewood Cliffs, New Jersey: Prentice-Hall, 1978.

[21] F. K. Soong and B. H. Juang, "Line spectrum pair (LSP) and speech data compression," *IEEE ICASSP*, vol. 1, pp. 1.10.1–4, May 1984.

[22] M. R. Schroeder, "Residual-excited LPC with vector quantization," *Speech Communication*, pp. 227–237, July 1988.

[23] M. R. Schroeder and B. S. Atal, "Code-excited linear prediction (CELP): High-quality speech at very low bit rates," *IEEE ICASSP*, pp. 937–940, 1985.

[24] I. A. Gerson and M. A. Jasuik, "Vector sum excited linear prediction (VSELP) speech coding at 8 kbps," *IEEE ICASSP*, vol. 1, pp. 461–464, 1990.

[25] B. Rele, "Simulation of VSELP Speech Encoder for mobile channels," Master's thesis, Virginia Polytechnic Institute and State University, Aug. 1993.

[26] Committee on Digital CDMA Cellular, "PN-3119 speech service option standard for wideband spread spectrum digital cellular system," tech. rep., EIA/TIA, TR45.5/93.0419.04, Apr. 1993.

[27] W. H. Press, W. T. Vetterling, S. A. Teukolsky, and B. P. Flannery, *Numerical Recipes in C: The Art of Scientific Computing.* New York: Cambridge University Press, second ed., 1992.

[28] Y. Li, "Bit Error Rate simulation of a CDMA system for Personal Communications," Master's thesis, Virginia Polytechnic Institute and State University, June 1993.

[29] J. Lichtenstein, "Low computational complexity Bit Error Rate simulation for personal communications systems in multipath and fading environments," Master's thesis, Virginia Polytechnic Institute and State University, Apr. 1994.

[30] N. Kitawaki, "Quality assessment of coded speech," in *Advances in Speech Signal Processing* (S. Furui and M. M. Sondhi, eds.), pp. 357–386, New York: Marcel Dekker, Inc., June 1992.

[31] A. V. Oppenheim and R. W. Schafer, *Discrete-Time Signal Processing.* Englewood Cliffs, New Jersey: Prentice Hall, 1989.

[32] J. D. Markel and A. H. Gray, *Linear Prediction of Speech.* Berlin: Springer-Verlag, 1976.

[33] R. E. Ziemer and R. L. Peterson, *Introduction to Digital Communications.* New York: Macmillan Publishing Company, 1992.

[34] P. E. Papamichalis, *Practical Approaches to Speech Coding.* Englewood Cliffs, New Jersey: Prentice Hall, 1987.

[35] R. W. Schafer and J. D. Markel, eds., *Speech Analysis.* New York: John Wiley & Sons, 1979.

[36] N. S. Jayant, "Adaptive delta modulation with a one-bit memory," *Bell System Technical Journal*, pp. 321–342, Mar. 1970.

[37] A. Gersho, "On the structure of vector quantizers," *IEEE Transactions on Information Theory*, vol. 28, pp. 157–166, Mar. 1982.

[38] G. D. Forney, "The Viterbi algorithm," *Proceedings of the IEEE*, vol. 61, pp. 268–278, Mar. 1973.

[39] "IEEE recommended practice for speech quality measurements," *IEEE Transactions on Audio and Electroacoustics*, pp. 227–246, Sept. 1969.

[40] R. L. Pickholtz, D. L. Schilling, and L. B. Milstein, "Theory of spread-spectrum communication — a tutorial," *IEEE Transactions on Communications*, vol. COM-30, pp. 855–884, May 1982.

[41] N. Kitawaki, K. Itoh, M. Honda, and K. Kakehi, "Comparison of objective speech quality measures for voiceband codecs," *IEEE ICASSP*, pp. 1000–1003, July 1982.

[42] P. Combescure, A. L. Guyader, and A. Gilloire, "Quality evaluation of 32 kb/s coded speech by means of degradation category ratings," *IEEE ICASSP*, pp. 988–991, July 1982.

[43] T. P. Barnwell and S. R. Quackenbush, "An analysis of objectively computable measures for speech quality testing," *IEEE ICASSP*, pp. 996–999, July 1982.

[44] B. S. Atal and L. S. Hanauer, "Speech analysis and synthesis by linear prediction of the speech wave," *Journal of the Acoustical Society of America*, vol. 50, pp. 637–655, 1971.

[45] M. R. Schroeder, "Vocoders:Analysis and synthesis of speech," *Proceedings of the IEEE*, vol. 54, pp. 720–734, May 1966.

[46] L. R. Rabiner, "On the use of autocorrelation analysis for pitch detection," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 26, pp. 24–33, Feb. 1977.

[47] M. Nakatsui and P. Mermelstein, "Subjective speech-to-noise ratio as a measure of speech quality for digital waveform coders," *Journal of the Acoustical Society of America*, vol. 72, pp. 1136–1144, Oct. 1982.

[48] J. Makhoul, "Linear prediction: A tutorial review," *Proceedings of the IEEE*, vol. 63, pp. 561–580, Apr. 1975.

[49] N. S. Jayant and P. Noll, "Speech coding with time-varying bit allocation to excitation and LPC parameters," *IEEE ICASSP*, vol. 1, pp. 65–68, 1989.

Appendix A

Execution Instructions

All the files are in the directory /home/u1/sharath/qcelp.

First the executable **qcelp** is created by typing out the following UNIX command.

make -f makefile

Then the executable **analysis** is created by typing out the following.

gcc -o analysis analysis.c adc.c random.c -I/usr/local/matlab/extern/include /usr/local/matlab/extern/lib/sun4/libmat.a

Then the program can be run by typing

analysis -i *audiofile_in* **-o** *audiofile_out* [**-f bbefile**] [**-p BER**]

Only one of **-f** and **-p** options must be given. If neither of these options is given, then just the vocoder is simulated.

VITA

Sharath Manjunath began his studies toward an M. S. degree at Virginia Polytechnic Institute and State University in August 1992. He joined the Mobile and Portable Radio Research Group in the month of May, 1993. He is a member of the IEEE Communications Society, Signal Processing Society and the Vehicular Technology Society. He is also a member of the honor society of Phi Kappa Phi. Sharath hails from Bangalore, India, having completed his B. S. degree at the Mangalore University, in June 1992.