Robust Linear Prediction Analysis for Low Bit-Rate Speech Coding

Nanda Prasetiyo Koestoer B. Eng (Hon) (1998)

School of Microelectronic Engineering
Faculty of Engineering and Information Technology
Griffith University
Brisbane, Australia

This dissertation is submitted in fulfilment of the requirements of the degree of Doctor of Philosophy

November 2002

Abstract

Speech coding is a very important area of research in digital signal processing. It is a fundamental element of digital communications and has progressed at a fast pace in parallel to the increase of demands in telecommunication services and capabilities.

Most of the speech coders reported in the literature are based on linear prediction (LP) analysis. Code Excited Linear Predictive (CELP) coder is a typical and popular example of this class of coders. This coder performs LP analysis of speech for extracting LP coefficients and employs an analysis-by-synthesis procedure to search a stochastic codebook to compute the excitation signal. The method used for performing LP analysis plays an important role in the design of a CELP coder. The autocorrelation method is conventionally used for LP analysis. Though this works reasonably well for noise-free (clean) speech, its performance goes down when signal is corrupted by noise.

Spectral analysis of speech signals in noisy environments is an aspect of speech coding that deserves more attention. This dissertation studies the application of recently proposed robust LP analysis methods for estimating the power spectrum envelope of speech signals. These methods are the moving average, moving maximum and average threshold methods. The proposed methods will be compared to the more commonly used methods of LP analysis, such as the conventional autocorrelation method and the Spectral Envelope Estimation Vocoder (SEEVOC) method.

The Linear Predictive Coding (LPC) spectrum calculated from these proposed methods are shown to be more robust. These methods work as well as the conventional methods when the speech signal is clean or has high signal-to-noise ratio.

Also, these robust methods give less quantisation distortion than the conventional methods. The application of these robust methods for speech compression using the CELP coder provides better speech quality when compared to the conventional LP analysis methods.

Acknowledgments

Firstly I wish to express my deepest gratitude to my supervisor Prof. Kuldip Paliwal for all the support and guidance he has offered me. I am very grateful for the knowledge and inspirational wisdom he has shared with me during the course of my study.

I am also thankful to the support I have received from the School of Microelectronic Engineering at Griffith University. The technical support and facilities have been essential in providing a great academic environment for me to complete my study. Specifically I would like to thank everyone at the Signal Processing Laboratory, with which I have had the honour of being associated with. The suggestions, discussions, valuable advice and support provided by the people associated with the laboratory, including visiting researchers, has been crucial during the progression of this work. Special mention goes to Brett Wildermoth, whose assistance in using the laboratory facilities was very beneficial to my research.

Very special thanks go to my closest friend, Shelley Kemp, whose support has been ever-present during my times of need. She is very dear to me and has been responsible for the best times of my life. Finally, I would like to thank my family for everything they have given me during this time. I will forever be grateful to experience their love and support.

Statement of Originality

This work has not previously been submitted for a degree or diploma in any university. To the best of my knowledge and belief, the thesis contains no material previously published or written by another person except where due reference is made in the thesis itself.

Nanda Prasetiyo Koestoer
November 2002

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Speech Coding

Speech coding has been a common area of research in signal processing since the introduction of wire-based telephones. Numerous speech coding techniques have been thoroughly researched and developed, spurned further by the advances in Internet technology and wireless communication [1]. Speech coding is a fundamental element of digital communications, continuously attracting attention due to the increase of demands in telecommunication services and capabilities. Application of speech coders for signal processing purposes has improved at a very fast pace throughout the years in order to allow it to take advantage of the increasing capabilities of communication technology infrastructure and computer hardware. Additional background information regarding the advances of speech coding in communication technology can be attained in [2], [3], [4] and [5].

This dissertation focuses on the area of speech coding. This particular area of research has become a fundamental necessity due to the bandwidth limitation of most signal transmission systems. Ideally in speech coding, a digital representation of a speech signal is coded using a minimum number of bits to achieve a satisfactory quality of the synthesised signal whilst maintaining a reasonable computational complexity.

Speech coding has two main applications: digital transmission and storage of speech signals. In speech coding, our aim is to minimise the bit-rate while preserving a certain quality of speech signal, or to improve speech quality at a certain bit-rate. In addition to these two attributes (bit-rate and speech quality), a speech coder has to concentrate on other attributes during its design. Importance of these attributes varies with the application to which the speech coder is used. For example, speech coders in general have the following attributes: bit-rate, speech quality, computational complexity, coder delay and sensitivity to channel errors. However, in broad terms the main goal in designing speech coders is to produce a naturally sounded reconstructed speech with low bit-rate and system cost.

Most speech coding methods have been designed to remove redundancies and irrelevant information contained in speech, thus aiming toward producing high quality speech with low bit-rates. The optimisation of the bit-rate and quality of the synthesised signal is closely related, where an improvement of one aspect compensates to the degradation of the other. Hence, the main development issue usually evolves around the compromise between the need for low rate digital representation of speech and the demand for high quality speech reconstruction.

Most of the speech coders reported in the literature are based on linear prediction (LP) analysis. A typical and popular example of this class of coders is the Code Excited Linear Predictive (CELP) coder. This Linear Predictive Coding (LPC) method performs LP analysis of speech for extracting LP parameters or coefficients and employs an analysis-by-synthesis procedure to search a stochastic codebook to compute the excitation signal. The autocorrelation method is conventionally used for LP analysis. Though this works reasonably well for clean speech, its performance deteriorates when signal is corrupted by noise.

The motivation behind this research is to introduce new methods of power spectrum envelope estimation for the LP analysis. In general, LP analysis has been used in the past in a number of applications such as speech coding, speech recognition and speaker recognition. Its most successful application is perhaps in speech coding where it is used to estimate the parameters of an all-pole model representing the envelope of the signal power spectrum [6]. It is highly beneficial to improve the performance of one of the most widely used time-frequency signal analysis in the speech compression field of research.

1.2 Research Objective

The objective of the research is to improve the robustness of the widely used LP analysis method of spectrum estimation in noisy environments. There has been a wide range of research and numerous publications regarding the performance of digital speech coding in real-life applications where undesirable noise is introduced to the system. Most research of signal processing in noisy conditions focuses on either the enhancement of speech, detection of pauses in speech, or noise cancellation, which are dependent or independent to the system. With the aim to achieve the same goal whilst improving LP analysis, a new method in estimating the envelope of the noise-corrupted signal's power spectrum is introduced.

An example of a speech frame affected by noise can be seen in Figure 1.1. It can be seen that as noise is introduced, the lower-level peaks of the power spectrum are affected most. Generally, noise affects the power spectrum of speech signal in 2 areas: a) the space between the harmonic peaks (Figure 1.1a shows the first few harmonic peaks, marked with circles) and b) the non-formant regions of the spectrum (area inside the box in Figure 1.1b). Because of this, the LPC spectrum of such a signal would be severely distorted, as it treats the high and low level peaks equally.

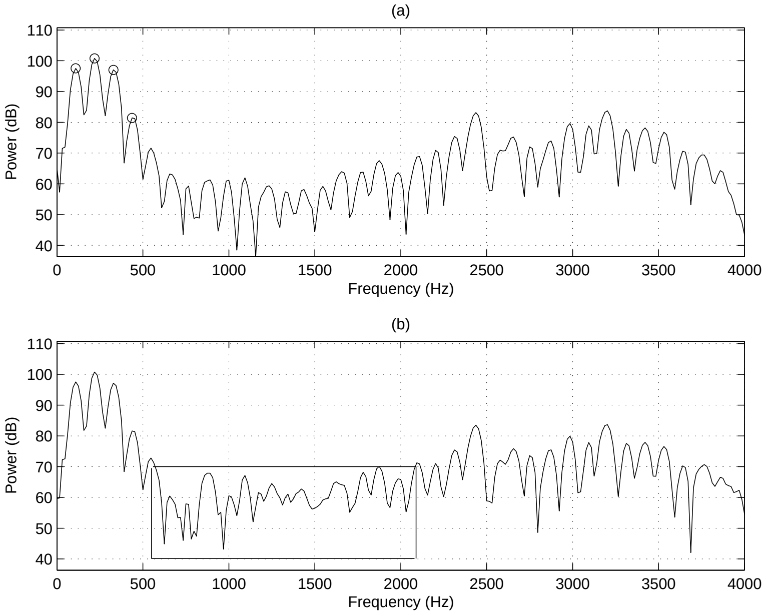

Figure 1.1: Power spectrum of speech for (a) clean signal (no noise) and (b) signal affected by noise (SNR=25 dB).

In order to overcome this problem, three new spectral envelope estimation methods are proposed; these are the moving average (MA), moving maximum (MM) and average threshold (AT) method. These methods rely more on the harmonics peaks and ignore valleys between the harmonic peaks. Hence when noise is introduced, the estimated envelope of the power spectrum would maintain the general shape of the power spectrum, whilst not being overly affected by the noise. These methods are designed to achieve: a) a more robust method for spectral analysis of signals introduced with real-world noise and b) better performance in terms of quantisation distortion for application in low bit-rate speech coders.

In this dissertation, simulation results are provided to show that the proposed methods present more robust methods of LP analysis when speech signal is affected by noise, without degrading its accuracy. In later chapters, the proposed methods are presented as applications in a low bit-rate compression scheme. Results relating to the quantisation performance of its LP parameters are included. It will be shown that quantisation of the LP parameters calculated using the robust methods performs better than quantisation of the LP parameters calculated using the conventional methods.

1.3 Thesis Organisation

A complete outline of the thesis is detailed as follows. Chapter 2 reviews the background theory of LP analysis and low bit-rate speech coding, specifically the Code Excited Linear Predictive (CELP) coder. The autocorrelation method of LP analysis is explained together with the SEEVOC method aimed at improving the performance of LP analysis. Quantisation of the LP parameters, covering the different LP parameter transformation methods, is also discussed in this chapter. Chapter 3 introduces the proposed methods of LP analysis, which includes explanation relating to the methodology and design of each proposed method. This chapter also investigates the robustness and accuracy of the proposed LP analysis methods in clean and noisy environments. A brief detail will also be included to discuss the speech database involved in these simulations. Chapter 4 investigates the quantisation of the LP parameters for the proposed methods and compares it to the conventional LP analysis methods. Chapter 5 investigates the application of low bit-rate speech coders using these robust methods. This thesis will conclude in Chapter 6, which includes a summary of this dissertation and future work.

Chapter 2

Speech Coding and Linear Prediction Analysis

2.1 Speech Production

Before studying the manipulation of digitised speech, it is crucial to have a basic understanding of how speech is produced. Speech is produced when the lungs force the direction of airflow to pass through the larynx into the vocal tract. In normal speech production, the air that is driven up from the lungs is passed through the glottis and vocal tract narrowing resulting in periodic or aperiodic (noise) excitation.

Parts of the mouth's anatomy, such as the jaw, tongue, lips, velum (soft palate) and nasal cavities, act as resonant cavities. These cavities modify the excitation spectrum that is emitted as vibrating sounds. Vowel sounds are produced with an open vocal tract with very little audible obstruction restricting the movement of air. Consonant sounds are produced with a relatively closed vocal tract, from temporary closure or narrowing of air passageway, resulting in high audible effect on the flow of air. A very basic model of speech production can be determined by approximating the individual processes of an excitation source, an acoustic filter (the vocal tract response) and the mouth characteristics during speech (Figure 2.1) [7].

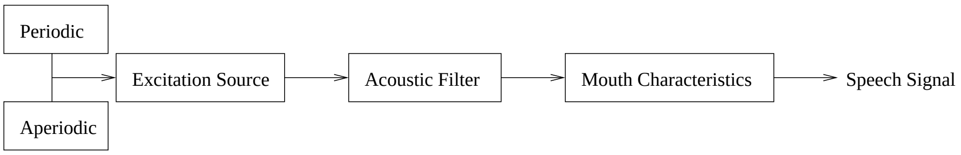

Figure 2.1: Basic speech production model.

2.2 Speech Signal

2.2.1 Time Domain Representation

Digital signal analysis of speech waves separates the speech into voiced (contains harmonic structure) and unvoiced speech (no harmonics structure, resembles white noise). For voiced speech, the opening and closing of the glottis results in a series of glottal pulses. This excitation possesses a periodic behaviour, where each glottal opening-and-closing cycle varies in shape and time period. A string of consecutive glottal pulses, also referred to as pitch pulses, results in a quasi-periodic excitation waveform.

An example of speech containing the word [*she*] can be seen in Figure 2.2. Unvoiced segments [*sh*] do not display any periodic behaviour, whereas the voiced segments [*e*] contain an obvious periodic behaviour in time domain.

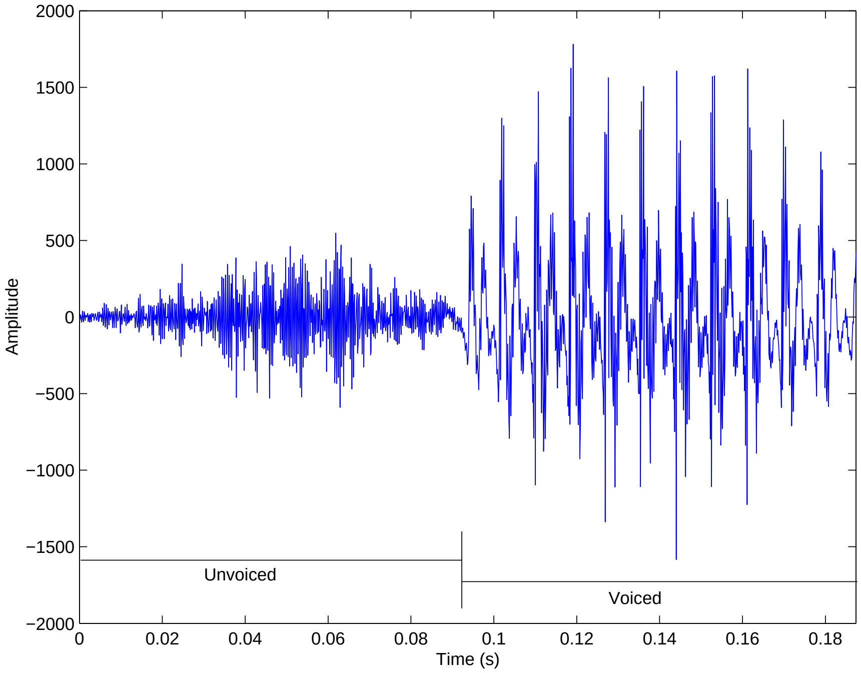

Figure 2.2: Speech signal [*she*] in time domain.

2.2.2 Frequency Domain Representation

In general it is understood that the vocal tract produces speech signals containing all-pole filter characteristics [8]. In speech perception, the human ear normally acts as a filter bank and classifies incoming signals into separate frequency components[1]. In parallel to the behaviour of the human speech perception system, discrete speech signals may be analysed in its frequency domain, where they are transformed into sinusoidal waves located at different frequencies simultaneously.

Figures 2.3a and 2.3b show the frequency domain of the segments that form the word [*she*]. The three spectrum plots of 20 ms from the unvoiced segment [*sh*] show no noticeable harmonic structure. Narrow spectral peaks can be observed

[1] *This is the general assumption of how the human perception system operates, it is not known for a fact that this case is completely accurate, however this generalisation has been deemed an accurate enough representation.*

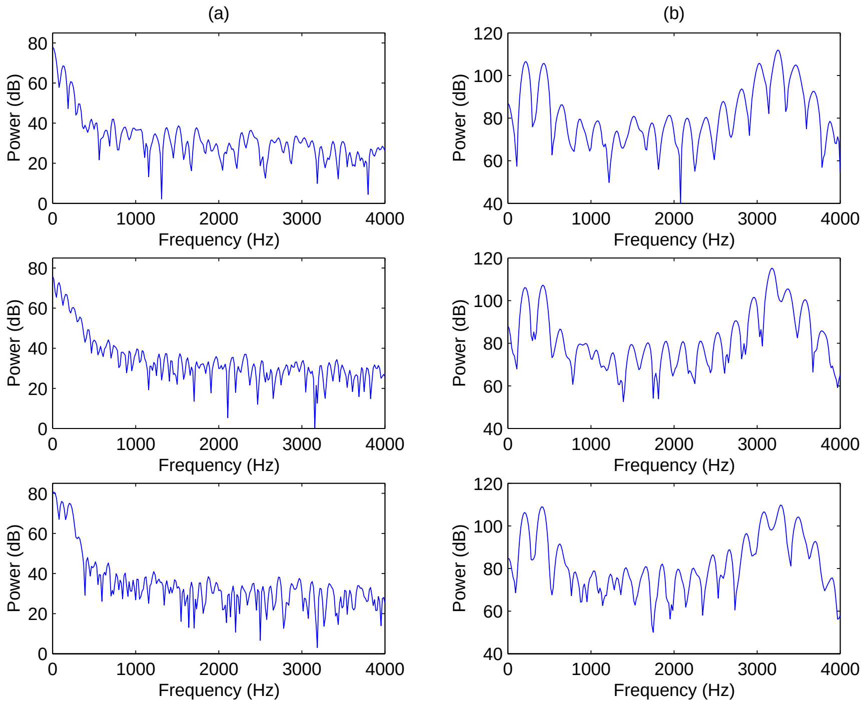

Figure 2.3: Speech signal [she] in frequency domain, (a) segments containing the unvoiced [sh] and (b) voiced [e] segments.

at periodic frequency intervals in the spectrum plots of the voiced segment [e]. This harmonic structure corresponds to the fundamental frequency of the glottis excitation.

Technically the human ear is capable of hearing signals ranging from 16 Hz to 18 kHz, depending on its amplitude. However it is known to be most sensitive for frequencies in the range of 1-5 kHz [9], hence distortion in the high frequency bandwidths is less noticeable to the human ear than distortion of equal amplitude in the low frequency areas. It should be noted that the increase of fundamental frequencies makes the signal less well defined by the more widely spaced harmonics. This is the contributing factor in the difficulty of analysing and sufficiently synthesising speech of a female or child in comparison to male speech[2].

2.3 Properties of Speech

The non-flat frequency response of the vocal tract provides correlation between neighbouring samples of the speech signal (short term correlation). It is also observed that during voiced speech, the periodic behaviour of the excitation results in the correlation between the corresponding samples of neighbouring pitch pulses (long term correlation).

A short-time window of samples (normally between 20-30 ms duration) is used to determine frequency domain properties of a signal segment. By assuming such segments to be stationary, its power spectrum is computed to represent its short-time spectral analysis. In the spectral domain, the short term correlation provides the envelope of its power spectrum, while the long term correlation provides the fine structure of the spectrum [10].

Voiced speech contains a harmonic structure in its power spectrum. As can be seen in Figure 2.4, the sharp spectral peaks are located at equal frequency intervals determined by its fundamental frequency. This explains the periodic structure of its time domain representation.

As mentioned in Section 1.1, bit-rate reduction is achieved by removing redundant information in speech data. Both correlations mentioned above introduce information redundancies in speech signal, which can be exploited using the LPC method of speech coding. LP analysis can be used to exploit the redundancies present in the short term correlation (as shown in Section 2.6).

[2] *It has been generally accepted that most male speech signals have a lower fundamental frequency than that of a female or child.*

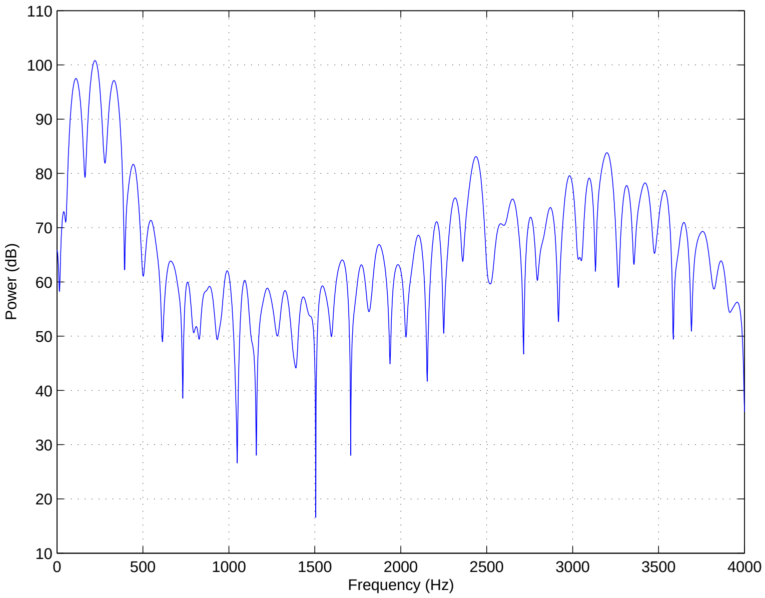

Figure 2.4: Power spectrum of speech segment [*e*] over 30 ms time frame.

Two main concerns in manipulating a speech segment are preservation of the speech content and transmission or storage convenience, in other words quality and size. The information content of speech should be easily extracted and synthesised from a speech encoding system. To produce comparable quality between the voiced and unvoiced speech, it would normally require less bits to encode the voiced speech than it would the unvoiced speech. This is due to the redundancies contained in the periodicity of the voiced speech, which can be further exploited.

2.4 Digital Encoding of Speech Signals

2.4.1 Sampling

Digital speech signals are speech waves recorded and sampled discretely for ease of use in communication technology. As the digital signal is a discrete representation of a continuous time signal sequence, it is necessary to represent it as mathematical functions of a continuous time variable t. Using a sampling period of T $(t = nT)$, the discrete-time signal can be represented as $x_{discrete}(n) = x_{analog}(nT)$.

Aliasing caused by the overlapping of high frequency on low frequency samples can be avoided by ensuring that the sampling frequency F_S is at least twice the maximum analog signal frequency F_N (known as the *Nyquist* frequency).

$$F_S \geq 2F_N \tag{2.1}$$

This dissertation focuses on telephone quality narrow-band speech, where analog signal is digitally sampled at 8 kHz. The conventional choice of sampling bit-rate for speech has been dictated by the telephone network capacity, band-limited between 300 and 3400 Hz. Phone lines normally attenuate frequencies above 3.2 kHz, allowing imperfect low pass filtering. This results in the common usage of speech signals with sampling frequency of 8 kHz and resolution of 16 bits/sample. Due to the direct progression from its early development with telephone communication technology, 8 kHz speech signals are still widely used in digital wireless or cellular communications. This standard of digitised speech has been deemed an adequate representation of the analog speech.

2.4.2 Quantisation

Quantisation is a popular application used in most signal compression methods. The methodology was developed for use in conventional communication technology. It was virtually impossible to transmit exact amplitudes of the signal and assuming that amplification on repeaters during transmission would not introduce noise or distortion to the signal. The same case holds for modern communication technology (i.e. wireless or broadband technology) where a desirable signal compression criterion may not be achieved by transmitting signal amplitudes of high precision. This is the reason behind applying only a certain number of discrete amplitude levels to represent the whole signal. This is more commonly referred to as quantisation.

The quantisation process is normally divided into two procedures: training and testing. The training procedure consists of an algorithm that processes a set of *codebook* samples and classifies them to a desired number of quantisation levels. The testing procedure then uses the quantisation levels to classify a set of input samples (separate from the codebook data used in the training procedure). As the quantisation levels are fixed discrete points, hence no further distortion is introduced to the data during transmission or compression. Therefore quantisation is one of the most important processes associated with discrete signal processing for digital transmission or storage purposes. When the signal from a quantisation process is received at the desired end, it is then decoded to form a series of reconstructed or synthesised samples, each having exact values as the original quantised signal before transmission.

Any alterations experienced during the compression of the signal are limited to the distortion created during the quantisation process, referred to as the *quantisation noise*. This noise is obtained when a singular signal or signal sequence is rounded to the nearest quantisation level.

For data compression purposes, the data that has been classified into the quantisation levels will then be represented by integer values associated with the respective levels. Signal distortion associated with analog signal transmission can then be avoided by using these discrete integer levels, therefore losing no information during the process. The operation of translating the sample points to desired integer levels has also the added benefit of decreasing the amount of data to transmit or store, albeit paying the price of degrading the accuracy of each signal point. A large number of quantisation methods have been developed throughout the years, but in general it can be based on two techniques: scalar and vector quantisation.

Scalar Quantisation

Scalar quantisation (SQ) is a technique developed to define the representation of a single signal sample with a single discrete value. Information contained in a string of signal samples can be compressed by representing it with distinctively less numbers of discrete values. The process associated with determining the quantisation levels has led to the introduction of quite a number of SQ methods, such as the uniformly spaced quantiser, adaptive quantisers, non-uniform quantisers (based on the logarithmic scale or the differential model), entropy-coded quantiser, etc.

Adaptive quantisers are SQ methods that adapts to the statistics of the quantiser input. Application of the LBG algorithm for SQ is a form of adaptive quantisation and will be explained in further detail in the next section.

Non-uniform quantisers, such as the Laplacian-distribution, γ-distribution, μ-law method and the optimum Gaussian-distribution technique[3], has been developed thoroughly and used widely through the years (further explanation regarding these methods can be obtained in [12], [13] and [14]).

Non-uniform quantisers that follow the log-scale behaviour are more commonly

[3] *Lloyd originally introduced this technique, commonly known as the* Lloyd-Max *quantiser, in 1957 and was further developed by Max in 1960 [11].*

used in speech signals, where the quantisation distortion of the higher-amplitude signals are usually masked by the louder signals. This in turn would leave the low-amplitude distortions to suffer more from noise than its larger counterpart. This particular behaviour of speech signals is what most quantisation processes in speech coding aim to exploit.

Another method of non-uniform quantisation is the *companded* quantiser. This method is based on expanding the region where the probability of the input occurring is high.

The most popular SQ technique, the Lloyd-Max non-uniform optimum scalar quantiser, approaches the design of quantising levels to be concentrated around the mean of the signal to compensate its *Gaussian* behaviour. This method is optimised with regards to the input signal's probability density function. This optimum scalar quantisation method, mainly used in speech coding, or signal compression in general, is normally embedded into the Pulse Code Modulation (PCM) technique, which is a time domain waveform encoding technique designed for digital data compression. This system is the basic method of producing a quantised version of an input signal for applications in signal transmission. For an N-bit transmission encoding system, each sample of the signal is quantised to one of the 2^N amplitude levels.

Spawning from this technique are the Differential-PCM (DPCM), which outputs a quantised version of the difference between the input signal and the predicted value of the input at each sample, and the Adaptive-DPCM (ADPCM), where its prediction coefficients and quantisation levels are varied depending on past reconstructed signals [15], [16], [17].

DPCM systems have an advantage of having a lower quantiser input RMS (*Root Mean Square*) value, thus needing fewer quantising levels to achieve minimum *mean-squared quantising error* (MSE). It should be noted here that these methods would still produce quantising noise; hence the aim is to minimise it accordingly.

PCM systems generally require more bandwidth and less power than the original signal. DPCM, and furthermore ADPCM, are more effective than PCM in usage for transmission or storage of digital signals. Despite that fact, PCM systems are used more commonly due to its possible usage in more general purposes [18]. This is much more beneficial when compared to DPCM system's dependency on signal characteristics [19].

There are also other time domain techniques developed in association with scalar quantisation, which include the Delta Modulation (DM) and Adaptive-DM (ADM). These methods are designed to develop correlation between adjacent samples. DM method of quantisation is basically a simplified form of the DPCM, where each quantiser bit is used in conjunction with a fixed first order predictor. ADM method of quantisation is developed to compensate the slope-overload distortion and granular noise problems associated with the DM technique [20].

Vector Quantisation

Background

The basic theory for this method of quantisation was first introduced by Shannon [21], and further developed as a theory of block source coding in [22], with regards to rate distortion theory. Prominent use of this theory was achieved when Linde, Buzo and Gray first introduced their vector quantisation algorithm (LBG algorithm) in [20]. The codebook design using the LBG algorithm is a clustering algorithm method also known as the generalised Lloyd's algorithm. Further research into this theory can also be seen in [23] from which its general design is prominently used in Chapters 4 and 5.

Vector quantisation (VQ), also known as the block or pattern matching quantisation, is a process executed when a set of signal values are quantised jointly as a single vector. It considers a number of samples as a block or vector and represents them for transmission as a single code. VQ offers a significant improvement in data compression algorithms where it minimises further the data storage required with respect to the methods used in SQ. The disadvantage of this quantisation method is that there is a significant increase in computational complexity during the analysis phase or training process. Database memory would also increase with the introduction of a larger size codebook. Despite its disadvantages, VQ remains a popular method of quantisation due to its improvements in encoding accuracy and transmission bit-rate.

VQ encoder maps a sequence of feature vectors to a digital symbol. These symbols indicate the identity of the closest vector to the input vector from the values obtained from a pre-calculated VQ dictionary or codebook. They are then transmitted as lower bit-rate representations of input vectors. The decoder process uses the transmit symbols as indexes into another copy of the codebook. Synthetic signal can then be calculated from the VQ symbols. This classification process may also be used in speech or speaker recognition systems.

Codebook Computation

The selection criterion of the codebook is the most defining part in designing an effective VQ coder. In determining the codebook, its vectors are trained to best represent the data samples, which are specifically designated for the VQ training procedure. The codebook computation procedure involves allocating a collection of vectors into what is referred to as *centroids*. These centroids represent the signal source and are designed to minimise the quantisation distortion across the synthesised signal.

The technique used in the design of the codebook, which will be used in the later chapters, is a combination of the full search codebook method and the LBG vector quantiser design. This is an exhaustive search, which compares the input vectors to every candidate vectors of the codebook. Quantisation distortion (D_m) is measured from the minimum MSE between the centroid C_m and the input vector x_i (data at the ith vector).

$$D_m = \frac{1}{M} \sum_{i=0}^{M-1} \left(\frac{1}{N} \sum_{k=0}^{N-1} d[x_{ik}, c_{mk}] \right) \tag{2.2}$$

where M is the number of input vectors classified to the centroid and N is the number of points in a vector.

For a B-bit VQ codebook, it would have 2^B number of codebook vectors. Each codebook vector is assigned to a codebook cell C_i (for $0 \leq i \leq (2^B - 1)$). The training procedure is defined as follows:

1. The first centroid (C_i at $i = 0$) is determined by averaging the entire input vectors. This vector consists of the average input vectors with the length of N (points in the vector), such that $C_i = [c_{i0},\, c_{i1},\, c_{i2},\, \ldots,\, c_{i(N-1)}]$.

2. C_i is then split into two close vectors, $C_i + \delta$ and $C_i - \delta$, where δ represents a small varying constant. These vectors are thus separated such that the new centroids can be optimised using the *mean* of the new vectors allocated to its cell.

3. The input vectors are then classified to the codebook cells by calculating its minimum distortion,

$$D_{m,i} = \frac{1}{N} \sum_{k=0}^{N-1} \min_{c \epsilon \alpha_m} d[x_{ik}, c] \tag{2.3}$$

   given $\alpha_m = C_i$; $i = 0,\, 1,\, \ldots,\, m-1$, and m is the current number of codebook cells.

4. Each centroid is recalculated during each iteration process by averaging the input vectors that are classified into each codebook cell.

5. Selection of centroids is considered optimum when D_m is minimised such that

$$\frac{(D_{m-1} - D_m)}{D_m} \leq \varepsilon \tag{2.4}$$

   where ε represents a fixed positive threshold. Optimum selection of centroids may be reached when no movements can be observed between the vectors used to form the centroids. If the centroids are not yet considered to be optimum, then the input vectors need to be reclassified (return to step 3).

6. The centroids are then split further (two vectors each) using δ and optimised also using the same algorithm as above (process repeats from step 3). This is consistent with the aim to continuously increment the codebook dimension depending on its allocated bits. These processes (steps 3 to 6) are repeated until the number of desired codebook vectors is achieved.

Computing the distortion of each cell and reconstructing the centroids globally will result in a minimised signal distortion. There are certain instances where the algorithm needs to complete a large number of iterations (number of repetition of steps 3-5) before reaching below its set threshold. In this case the distortion is deemed to reach its global minimum when a pre-defined number of iterations has been completed during the process. Although this approach is sub-optimal, it is deemed to be an efficient, yet still highly effective, method of VQ training.

VQ Designs

There are a number of different methods in designing a VQ codebook that has been developed throughout the years in order to produce optimum quantisation results. These methods are specifically designed to fulfil certain goals or achieve specific means.

Multistage VQ employs two or more VQ's consecutively, where each stage codes the error of its preceding stage. Split VQ separates the input signal into two or more sub-vectors, with each sub-vector coded with different VQ classes. Gain shape quantiser is a system where VQ, which is used to code the data vectors, is used in conjunction with SQ, which is used to code the vector lengths. Tree-structured VQ partitions the quantiser output to reduce its computational load. The cascaded likelihood VQ, as proposed in [24], is a sub-optimal vector coding method specifically designed for use with CELP systems normally operating at

4.8 kbps. Other methods, to name a few, include the lattice VQ, transform VQ, product code VQ, trellis VQ and hierarchical VQ (please refer to [23] and [25]).

As the original design of VQ is complex and computationally expensive, most of the methods mentioned above are aimed to trim the complexity, in some cases degrading the performance quality. Although SQ is still used in certain areas of signal coding, VQ is generally applied to most quantisation designs due to its importance in reducing the compression bit-rate.

2.5 Overview of Speech Coding Methods

2.5.1 Introduction

The main objective in compressing a digital signal is to represent information associated to the signal as economical as possible whilst retaining parameters sufficient to reconstruct the original signal. Reduction of data storage space or digital transmission rate should be balanced with the maximisation of synthesised signal quality, which is to preserve its intelligibility and naturalness for speech signals, whilst eliminating redundant signal information.

Numerous methods of speech coding have been developed to achieve the goals stated above. However as the dissertation is focused on the improvements proposed for LP analysis, thus the compression methods discussed here are the methods related to LPC design.

The LPC scheme is a common technique used for lossy data compression in signal processing. This method takes an analysis-by-synthesis approach where it extracts the needed parameters of a signal by minimising the error of the decoder output.

In extracting the parameters from the signal, the input must be driven in order to model the signal sequence. During the analysis stage, the signal's short-term correlation is determined using the LP analysis method. The long-term correlation of the signal is determined using *pitch prediction* to exploit the periodicity of the signal. The extracted prediction parameters are then transmitted and used in the signal reconstruction process at the synthesis stage.

LPC-10 is an early LPC design that employs the use of fixed excitation signals to drive the input signal (Section 2.5.2). The input signal may also be driven by a string of impulses, which is provided by an excitation generator. This LPC method is commonly referred to as the Multipulse Linear Predictive Coding (Section 2.5.3), which led to the development of the Code Excited Linear Prediction (CELP) coder (Section 2.5.4).

2.5.2 LPC-10

This method was developed based on the *channel vocoder* method[4]. The vocal tract filter of the input signal is modelled by a single linear filter as oppose to the use of a bank of filters in the channel vocoder. Synthesised speech can be modelled from the input signal using either random noise or periodic pulse generator (please refer to Figure 2.5).

The 2.4 kbit US Government Standard LPC-10 is the most widely used standard for this method, where an 8 kHz speech signal is divided into frames of 180 samples (frame length of 22.5 ms). This method has been documented to suffer in noisy environments [26], whilst suffering from poor sound quality due to the use of only two excitation signals.

[4] *This method is a conventional analysis-by-synthesis method of speech compression developed in the late 1930's [Dudley, 1939].*

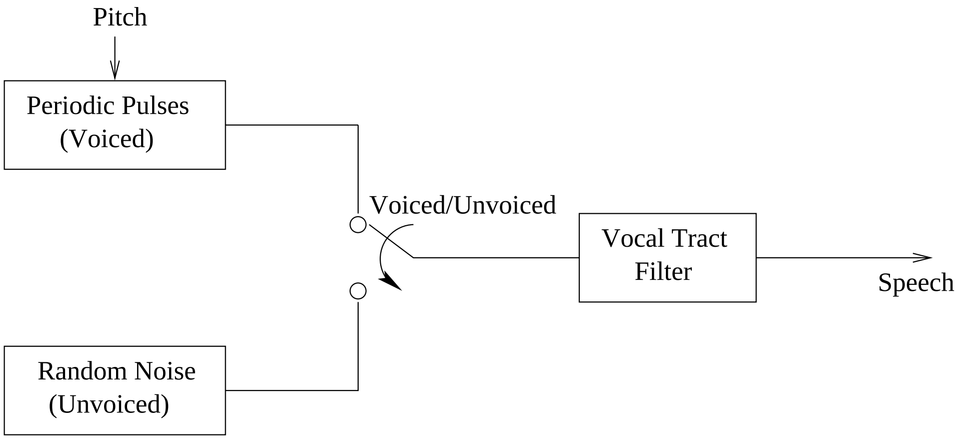

Figure 2.5: Basic speech synthesis model of the LPC-10 method.

2.5.3 Multipulse LPC

In this method of LPC, a stream of signals is modelled as the output of an all-pole filter, driven by an excitation function. As the title of this compression scheme indicates, the excitation function consists of a pulse sequence containing a small number of pulses, defined by their location and amplitude. Atal and Remde first introduced this multipulse excitation approach of LPC in [27]. A detailed discussion of the multipulse LPC is presented here as this method initiated the development of the CELP coder, which is prominently used throughout this dissertation.

A sequence of excitation pulses is computed for each frame of the signal. Increasing the number of excitation pulses would gradually improve the quality of the synthesised signal. However a minimised number of pulses will be needed to ensure an acceptable synthesised signal quality with an optimum compression ratio. It has been shown in [7] that only a small number of pulses (4 to 10 pulses) for each sub-frame are enough to produce an acceptable synthesised signal. Commonly a setting of 8 pulses per cluster of 64 samples is sufficient in generating the desired

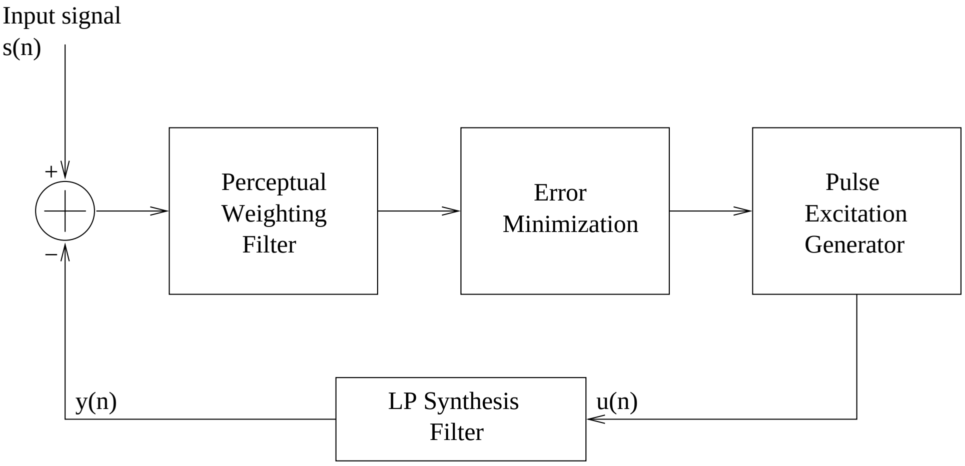

Figure 2.6: Block diagram of the multipulse coder.

input or residual signal with minimised distortion[5] [28].

The main focus in the design of this compression scheme is in determining the location and amplitude of the pulses. These pulses should closely represent the actual signal after being fed through a weighting filter. Excitations for the all-pole filter (or pole-zero filter, depending on its application) are created via an excitation generator that produces a sequence of pulses at certain locations and amplitudes. An LP synthesis filter is used to produce the synthetic signal waveform from the pulses.

Using an analysis-by-synthesis approach, the pulse locations and amplitudes are determined by minimising the weighted mean-squared error created by the difference between the original and the LP synthesis filtered signal. Each pulse determination process assumes that previous pulse amplitudes and locations are constant throughout the search. Although this may not be the most accurate manner in calculating the pulses, however it is deemed computationally efficient without much degrada-

[5] *For a signal with a sampling frequency of 8 kHz, with 20 ms frame sizes (160 samples) and an update rate of 4 updates per frame (each frame divided into 4 sub-frames of 5 ms segments), 5 pulses are generally used for each sub-frame.*

tion of accuracy. For m number of pulses and a frame length of N, an exhaustive search, which involves calculating every possibility of the pulses simultaneously, would need approximately N^m points of computation (depending on estimation methodology) in comparison to the chosen manner, which would only need $N \times m$ computation points.

Pulse Computation

The information content of each pulse contains of two values, its amplitude (β_k) and location (denoted by its position in the frame). Each pulse location number, referred to as n_k for every k^{th} pulse, can be seen in (2.5). The combination of pulses can be collectively defined as

$$u(n) = \sum_{k=0}^{m-1} \beta_k \delta(n - n_k) \tag{2.5}$$

where m is the number of pulses and δ_n is the *Kronecker* delta. Referring back to Figure 2.6, the signal $y(n)$ is obtained by weighting the pulse $u(n)$ with an impulse response $h(n)$, such that from

$$y(n) = u(n) \times h(n) \tag{2.6}$$

we get

$$y(n) = \sum_{k=0}^{m-1} \beta_k h(n - n_k) \tag{2.7}$$

Observing from Singhal and Atal [29], the squared error (E) must be minimised with respect to the pulse amplitudes and locations. Optimum pulse locations are determined by calculating the minimum error for all the possible locations and its optimum amplitudes in a set sub-frame [30].

$$E = \sum_{n=0}^{N-1} [s(n) - \beta_k h(n - n_k)]^2 \tag{2.8}$$

for N denoting length of the sub-frame. Solving for

$$\frac{\partial E}{\partial \beta_k} = 0, \tag{2.9}$$

we get

$$\beta_k = \frac{\sum_{n=0}^{N-1} s(n)h(n-n_k)}{\sum_{n=0}^{N-1} \left[h(n-n_k)\right]^2} \tag{2.10}$$

Substituting β_k back into E,

$$E = \sum_{n=0}^{N-1} s^2(n) - \frac{\sum_{n=0}^{N-1} \left[s(n)h(n-n_k)\right]^2}{\sum_{n=0}^{N-1} \left[h(n-n_k)\right]^2} \tag{2.11}$$

As $s(n)$ is the original signal, the second term of the equation would then have to be maximised. This introduces the autocorrelation (α) and cross-correlation (c) constants, where

$$\alpha(n_k) = \sum_{n=0}^{N-1} h^2(n-n_k) \tag{2.12}$$

and

$$c(n_k) = \sum_{n=0}^{N-1} s(n)h(n-n_k) \tag{2.13}$$

Pitch Prediction

In linear prediction, there is a period of underlying harmonic called the pitch period. In general, a transmitter system needs to estimate these pitch prediction coefficients in order to obtain a better representation of the signal. This information would also need to be transmitted together with the pulse data.

It has been well understood that the human ear is highly sensitive to pitch errors [31]. This has brought forth the development of more accurate pitch detection algorithms. The technique used here employs the autocorrelation (2.12) and cross-correlation (2.13) functions. This autocorrelation function provides a suitable approach in predicting the pitch period of the signal. This function should have a maximum value at each pitch period points. A pre-determined maximum coefficient is needed to help establish the pitch coefficient. The pitch coefficient is deemed to be reached when the autocorrelation value is larger than the set threshold.

2.5.4 CELP

This LPC method is a very common scheme used for low bit-rate data compression. This lossy scheme has been developed since the early to mid 1980's (formally introduced by Atal and Schroeder in [32]), but it has been used prominently only recently. A collection of random excitation signals is used to drive the vocal tract filter instead of using a codebook of pulse patterns.

Currently the CELP coder, as a signal compression technique, is very widely used for speech coding applications. This coder directly supersedes the development of the multipulse LPC. In general, the CELP coder has been specifically developed for compression of signals at bit-rate of 4.8 kbps, which is ideal for speech data. A detailed explanation regarding the operation of this coder is covered in Section 2.7.

The methods discussed in this section and in the two previous sections are very popular LPC techniques for application in speech coding. However as the focus of this dissertation is not on the design of the speech coders, then only one method is used to perform the speech coding simulations applied in later chapters. The CELP coder is selected over the previous two methods, as it is one of the most commonly researched and implemented speech coding methods and is deemed to provide the best quality sounding speech for low bit-rate compression [26].

2.6 LP Analysis

2.6.1 Background Theory

LP analysis of speech is historically one of the most important, and currently the most popular, speech analysis technique for low bit-rate speech coding. It was initially developed in the late 1960's [Atal and Schroeder, 1967] and further studied in the early 1970's with multiple publications ([6], [33], [34] and [35]) researching this theory. The theory and algorithms surrounding this matter have matured to the point where they are now an integral part of many real-world adaptive systems.

The generation of each phoneme during speech production is dependent on two factors: source excitation and the vocal tract shape. Modelling these two factors, assumed to be independent of each other, is crucial in modelling the speech production system. Ideally the vocal tract filter is modelled by a discrete time glottal excitation signal. LP analysis in speech coding is aimed at modelling the vocal tract model.

Referring to the name linear prediction itself, this method is based on the theorem that every predicted value is a result of a linear combination of its past values. The most popular vocal tract model for LP analysis is the autoregressive (AR) model. This analysis is a parametric method designed for discrete-time linear stochastic processes. It is based on a source-filter AR model where the vocal tract filter is modelled to be an all-pole linear filter.

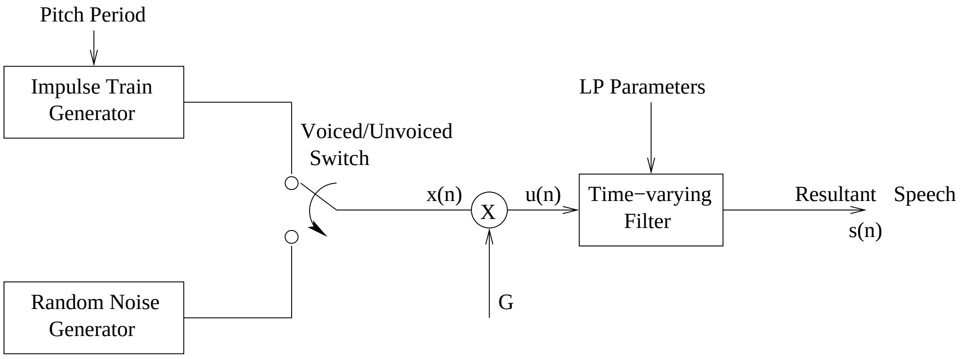

Figure 2.7: Speech processing model in LP analysis.

Application of LP analysis on speech signals can be used to estimate the basic speech parameters, such as pitch, formants, spectra and vocal tract area functions, which in turn can be applied for compression purposes. Speech signals can be modelled as an output of a linear time-varying system excited by pulses that are quasi-periodic (voiced) or random noise (unvoiced). Referring to Section 2.5.2 and Figure 2.7, the process of generating synthetic speech in LPC is designed to model such systems accurately.

The distinct advantage of this method is its ability to estimate the important speech parameters with a reasonable degree of accuracy and efficient computational speed. This is possible due to the amount of research devoted to this topic [36], [37], [38]. LP parameters, also referred to as LP coefficients, are determined from a finite sequence of samples by minimising the MSE between the original and its predicted signal. An example of a standard open-loop LP model can be seen in Figure 2.8.

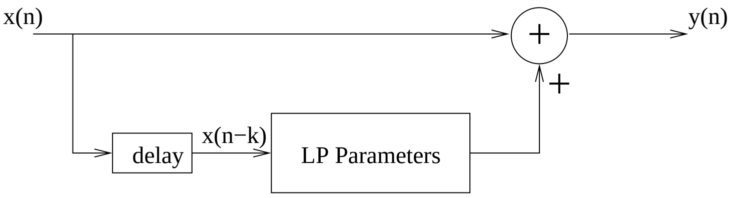

Figure 2.8: Open-loop AR model.

A mathematical representation of the open-loop LP model can be seen as follows,

$$y(n) = x(n) + \Sigma_{k=1}^{N} a_k x(n-k) \tag{2.14}$$

where N is the LP order and a_k is the LP parameters.

The resultant error $[e(n)]$ between the actual signal $[x(n)]$ and the linearly predicted signal $[y(n)]$ is what is quantised for transmission purposes in signal compression schemes. The filter involved in the determination of $e(n)$ computes its predicted results from a FIR filter. As the LP model becomes more accurate, the decrease in the quantisation error provides far less distortion in providing the synthesised signal. The difference between the true signal and the synthesised signal is called the LP residual or prediction error.

A more adaptive closed-loop model takes into account the quantised error signal into the all-pole filter equation.

$$\bar{x}(n) = \bar{e}(n) - \Sigma_{k=1}^{N} a_k \bar{x}(n-k) \tag{2.15}$$

For a closed-loop model, $\bar{e}(n)$ is the quantised version of $e(n)$. This quantised error signal is used at the receiver end as an excitation signal to the LP synthesis filter to compute the synthesised signal $\bar{x}(n)$.

Another approach used in LP analysis is the ARMA (autoregressive moving average) model. It is the most accurate approach for modelling the vocal tract [39]. It uses a linear combination of its past outputs in addition to a combination of its present and past inputs. This pole-zero model can only be derived by solving a set of non-linear equations. Obviously it is computationally more efficient to solve only one set of linear equations in an all-pole model. This is the reason why the all-pole (AR) model is the most commonly used model in LP analysis. The zeros, which arise in unvoiced and nasal sounds, are thus approximately modelled by poles.

It has also been determined that the human perception system is more sensitive to spectral poles than zeros, which is another reason why the AR model is popular for use in speech coding [40]. This leads to the most obvious deficiency of this model where it assumes speech spectra to be a perfect all-pole model without any zero present. Nevertheless, the information gathered by the all-pole filters has been deemed quite successful in predicting a sample as a weighted sum of past samples.

Limitations of the LP Analysis

Most signals in real life show a non-stationary behaviour. The main assumption involved in LP analysis is that for a narrow finite time frame the signal behaviour is stationary. This is still known to be the main disadvantage of the LP analysis method.

The methodology of the LP analysis involves describing large records of data by a uniform set of parameters containing its significant process information. Hence, in speech coding, a set of information computed from LP analysis (LP parameters) may represent a considerably larger set of speech data. However computing a set of LP parameters that accurately model the vocal tract filter would be compromised by its order of analysis (LP order). Clearly a higher LP order would constitute to a more accurate LP analysis.

Another limitation of LP analysis is introduced by data windowing. The choice of windowing function will always present a trade-off between time and frequency resolution (please refer to page 32). Despite these restrictions, spectral estimation via LP analysis remains very popular because it still provides a very good frequency and time resolution, and its ease of application for signal compression.

Spectral Analysis

The human auditory system is known to perform spectral analysis upon speech signal, hence the original motivation in analysing speech in its frequency domain [41], [42]. The vocal tract produces signals that are more precise and consistent in frequency domain than in time domain [43].

Spectral analysis applied for use in speech coding examines the behaviour of speech mainly in its frequency domain by determining relative magnitudes of the different harmonics of the speech signal. The spectral envelope estimate of a power spectrum calculated using LP analysis for use in LPC is referred to as the LPC spectrum. The LPC spectrum offers a concise representation of important signal properties, which largely simplifies the control of synthesis models.

In processing a digital signal in its frequency domain, the quality of a synthetic signal relies heavily on how well its LPC spectrum is estimated. Fulfilling the spectral envelope properties is the main requirement need to be reached in developing a spectrum estimation method. These properties are listed below:

- *Robust.* Spectral envelopes should maintain its general shape and characteristics when introduced to varying environments.
- *Envelope shape fitting.* Envelope should match the shape associated with the spectral peaks as close fitting as possible, following the link between the sinusoidal peaks.
- *Smoothness.* General shape of the signal magnitude distribution over frequency should be easily achieved with no sudden fluctuation in the envelope.
- *Acclimatisation.* Spectral envelope should accurately follow the sudden variations between two consecutive short-time spectral segments.

The importance in performing an accurate analysis of the power spectral can be seen via the rigorous research devoted to this topic in general [44], [45], [46], with regards to its robustness (Section 2.6.3) and most importantly in the study of the AR model [47], [48] and discrete Fourier transform [49].

In order to attain specific characteristics of the power spectrum, each frame of a speech signal can be parameterised. The source-filter model of speech production is generally applied as a theoretical model in the speech processing analysis. It is used to model the physiology of the human speech system. By modelling an all-pole filter on the resonances of the speech spectrum, filter coefficients can be obtained. These filter coefficients are what is used to estimate the power spectrum of a speech frame. These coefficients can be further quantised via a conversion to spectral parameters for applications in signal compression. This analysis is achieved via the Fourier analysis.

Fourier Analysis

As developed by Jean Baptiste Joseph Fourier in 1807, the underlying theory behind the development of the Fourier analysis is that a set of sinusoidal waves at separate frequency locations can be used to represent a discrete time length of a speech signal. The reverse process has been shown to be true in a mathematical sense [50].

It is very common for information to be encoded in the sinusoidal waveforms that form a signal. Specifically for spectral analysis, the shape of the waveform in its time domain is not important, as the key spectral information consist of its amplitude, phase and frequency of the sinusoidal representations. Any waveform that is assumed to be periodic can be analysed as a combination of the above mentioned harmonically related exponents. The Fourier transform of speech signals provide both spectral magnitude and phase with respect to its frequency. The importance of the Fourier transform in LP analysis is in the design of the power spectrum, whilst the phase spectrum is relatively unimportant perceptually [51].

In LP analysis, a short-time Fourier transform is used to represent the time-varying properties of a waveform in the frequency domain. The Fourier transform is defined as follows,

$$X_k(f) = \sum_{n=-\infty}^{\infty} w(k-n)x(n)e^{(-j2\pi fn)} \tag{2.16}$$

The window function $w(k-n)$ is a real window sequence used to isolate the portion of the input sequence that will be analysed at a particular time index k.

An ideal window function would acquire a frequency response with a narrow main lobe, which increases resolution, and no side lobes, which dictates the frequency leakage. The rectangular window function separates the signal into finite-sized frames without introducing any weighting. This introduces oscillation at the points of discontinuity, known as the *Gibbs phenomenon*.

Many window functions have been generated to improve upon the basic rectangular window design, such as hamming, hanning, bartlett, blackman, kaiser, etc., each having different specification with regards to its frequency response. In this dissertation, LP analysis was performed on frames weighted with the hamming window. This window, $w(n)$ in (2.17), was chosen as it provides a good balance between its mainlobe width and sidelobe attenuation.

$$w(n) = \begin{cases} 0.54 - 0.46cos(\frac{2\pi n}{N}) & ; 0 \leq n \leq N-1 \\ 0 & ; \text{otherwise} \end{cases} \tag{2.17}$$

The hamming window is also deemed to be adequate in determining the accuracy for approximating the transfer function of the vocal tract. This is a crucial aspect when calculating reflection coefficients for quantisation purposes (Section 2.8.2) [52].

In speech coding, a frame length of 20 to 30 ms is commonly chosen for LP analysis. Speech samples are assumed to be stationary for that period of time. This introduces the use of Discrete Fourier Transform (DFT) for applications in discrete systems.

The DFT is a widely used analogy for time-to-frequency transformation and is the central algorithm in most spectrum analysis systems. It is defined in mathematical terms as follows,

$$X(f) = \sum_{n=0}^{N-1} x(n)e^{(-j2\pi fn)} \tag{2.18}$$

Large discrete frame lengths give poor time resolution but good frequency resolution. As an example, in the speech coder used in Chapter 5, the DFT bandwidth is chosen to be 50 Hz. Hence an 8 kHz discrete signal constitutes to a frame length (N) of 160 samples.

DFT coefficients computed from a finite duration of samples are values of the z-transform at periodically spaced locations around the unity circle. It constitutes a unique representation of that particular sequence. Although the DFT and Inverse-DFT (IDFT) relations are developed based on its periodic sequences, it does have the ability to represent a finite duration of samples.

Application of DFT in real-world systems however is still computationally costly, especially for a long stream of discrete signals. This brought forth the development of the Fast Fourier Transform (FFT). FFT is a form of Fourier transform that is aimed purely to reduce the computation time required for DFT. The most widely used FFT algorithm is the radix-2 decimation-in-time and decimation-in-frequency method [10].

Referring back to Figures 1.1 and 2.4, the frequency components in the power spectrum computed via the Fourier analysis can be observed. The lowest frequency component of the frequency domain is known as the fundamental frequency, while the others are known to be harmonic frequencies. Usually the harmonics are represented with a number corresponding to the multiple of the fundamental frequency. In an ideal periodic signal, the harmonics would be located exactly at fundamental frequencies apart from each other. In a musical sense, the harmonics numbered by powers of two represents the octave levels.

Non-periodic waveforms may also be analysed by Fourier means which results in a complex integral. Disregarding phase, a spectral analysis can still be generated where the frequency components are not exact multiples of the fundamental frequencies. This would create difficulties during LP analysis, as the further away the spectral analysis resembles a harmonic model, the harder it is to perceive pitch in the signal.

When complex waveforms are introduced (for example, speech signals affected by noise), the Fourier analysis normally gives a statistical answer. It is very possible to locate a particular frequency component over a large time frame; however allocating constant amplitude would remain a problem. The probability curve is then what is normally obtained for the spectral analysis. A narrow-band noise will appear to be similar to a pitched tone, but for the complete bandwidth the distinction of the pitch tends to disappear (please refer to Figure 2.3).

LPC Spectrum

Quality of a decoded synthetic signal via LP analysis depends heavily on how well the spectral envelope is estimated. The accuracy of the estimation defines how sufficient the signal properties are captured. The power spectrum, also known as Power Spectral Density (PSD), is plainly a mathematical representation of amount of power as a function of frequency. In a mathematical sense, the PSD (symbolically defined as $P_{xx}(f)$) is defined

$$P_{xx}(f) = \sum_{k=-\infty}^{\infty} R_{xx}(k)e^{(-j2\pi fk)} \tag{2.19}$$

where $R_{xx}(k)$ is the autocorrelation function of an input signal. As can be seen, PSD is defined as a Fourier transform of the autocorrelation sequence in its time series. In LP analysis, the PSD is normally computed using the periodogram method of spectrum estimation.

Periodogram Spectrum

Periodogram method of spectrum estimation, or also known as the sample spectrum method, is the classic way of estimating the power spectrum [Schuster, 1898], [53]. It was originally designed to observe the *hidden* periodicities in the data.

This method is described using a direct computation of the squared constant multiplier of the Fourier transform of its time series. The periodogram spectra $Per_{xx}(f)$ is based on a direct approach through a Fourier transform on a frame of data, generally performed through the Fast Fourier Transform (FFT).

$$Per_{xx}(f) = \frac{1}{N}\left|\sum_{n=0}^{N-1} x(n)e^{(-j2\pi fn)}\right|^2 = \frac{1}{N}\left|X(f)\right|^2 \tag{2.20}$$

Periodogram spectrum is a direct realisation of the PSD, hence as $N \rightarrow \infty$, then $Per_{xx}(f)$ should ideally resemble $P_{xx}(f)$ (2.19). However this is not the case, where the periodogram is not a true representation of the PSD. Spectral leakage is encountered on the spectrum, which is due to the limitation of short-time analysis on finite-length data. However, as explained by Kay and Marple [54], a statistically consistent spectrum using this method can only be reached by separating the data sequence into smoothed segments. Leakage effects that occur due to data windowing can be minimised through the selection of non-uniform weighted windows (please refer to page 32 for selection of weighting window).

Despite the limitation stated above and its high vulnerability to noise, the periodogram spectrum still provides a very good basis to be used in designing a robust spectral envelope estimation method. All the proposed methods to estimate the power spectrum envelope in Chapter 3 are designed to manipulate the periodogram spectrum of a given signal. This is achieved in order to produce a finite-length set of filter coefficients that could best map its spectrum shape and characteristics. This approach is commonly used due to the computational efficiency of the FFT algorithm.

Bandwidth Widening

Synthesis filters with sharp spectral peaks are known to exist in LP analysis. A slight expansion of the bandwidth is normally applied to avoid such sharp spectral peaks. The bandwidth of the formant peaks would need to be widened in the frequency response using the methodology set out below.

The bandwidth widening (may also be referred to using the terminology bandwidth broadening) coefficient, γ, can be weighed upon the roots of the all-pole filter model $H(z)$ such that the new filter model $\acute{H}(z)$ can be described as follows,

$$\acute{H}(z) = \frac{1}{A(\gamma z)} \tag{2.21}$$

Its prediction coefficients can then be calculated as follows,

$$\acute{a}_k = a_k \gamma^k \tag{2.22}$$

where $1 \leq k \leq p$ and p denotes its LP order.

In speech analysis, bandwidth widening is normally set in the range of 10-25 Hz. The coefficient γ can be calculated from the desired expansion factor F_{bw} as follows,

$$\gamma = e^{\frac{-F_{bw}\pi}{F_s}} \tag{2.23}$$

for sampling frequency F_s.

2.6.2 Conventional LP Analysis Methods

There are a number of methods that can be used in LP analysis to obtain the LP parameters. These methods are the lattice, covariance and the most commonly used autocorrelation methods.

Lattice Method

This method incorporates the application of a forward and backward predictor. Although this method guarantees a stable filter, a considerable amount of storage space is needed to fulfil the computation process. A popular implementation of this method for LP analysis by Burg [6], also referred to as the Burg method, has a disadvantage in its line splitting tendencies and the dependency of the peak locations on phase. There is also the Recursive Maximum Likelihood Estimation (RMLE) method, which is similar to the Burg method in a way that it maximises its likelihood functions as opposed to minimising its prediction error [31].

Covariance Method

Covariance method is an algorithm specified for frame-to-frame basis estimation, with sets of data limited to $0 \leq n \leq N-1$ interval. This method windows the residual error signal rather than the actual signal in a way to minimise its error. For an LP order of p, the signal is assumed to be known for the set of values $-p \leq n \leq N-1$. No values outside this interval are needed for computation. This results in a covariance matrix solution that would be symmetric, however it should be noted that it is not a *Toeplitz*[6] matrix.

Although this method guarantees stability in most cases, the inversion of the covariance matrix is computationally expensive. This is the reason behind the selection of the autocorrelation method for LP analysis in this dissertation.

Autocorrelation Method

Autocorrelation method of LP analysis, or also referred to as the *Wiener-Khintchine theorem*, is the most popular method of short-term LP analysis. This method provides the most computationally efficient manner in determining the LP parameters with guaranteed stability. It takes advantage of the Toeplitz property possessed by the autocorrelation matrix.

The autocorrelation function of a signal is the inverse Fourier transform of its power spectrum. This function represents the correlation between adjacent signal samples. It measures the similarities between the current signal $x(n)$ with its past values as a function of time. The autocorrelation function $[R_{xx}(k)]$ is defined

$$R_{xx}(k) = \varepsilon[x(n)x(n+k)] \tag{2.24}$$

where ε is the expectation operator, as can also be seen in (2.28).

[6] *The Toeplitz matrix is a symmetric matrix where all elements along any given diagonal are equal.*

In matrix form, the autocorrelation function can be represented as follows,

$$\begin{bmatrix} R_{xx}(0) & R_{xx}(1) & R_{xx}(2) & \dots & R_{xx}(p-1) \\ R_{xx}(1) & R_{xx}(0) & R_{xx}(1) & \dots & R_{xx}(p-2) \\ R_{xx}(2) & R_{xx}(1) & R_{xx}(0) & \dots & R_{xx}(p-3) \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ R_{xx}(p-1) & R_{xx}(p-2) & R_{xx}(p-3) & \dots & R_{xx}(0) \end{bmatrix} \begin{bmatrix} \alpha_1 \\ \alpha_2 \\ \alpha_3 \\ \vdots \\ \alpha_p \end{bmatrix} = \begin{bmatrix} R_{xx}(1) \\ R_{xx}(2) \\ R_{xx}(3) \\ \vdots \\ R_{xx}(p) \end{bmatrix} \quad (2.25)$$

where p symbolises the LP order and α_k the LP parameters. The series of linear equations above is commonly referred to as the Yule-Walker equation. The minimum mean square prediction error E can thus be obtained as follows,

$$E = R_{xx}(0) - \sum_{k=1}^{p} \alpha_k R_{xx}(k) \quad (2.26)$$

Assuming that the process is stationary, the corresponding autocorrelation vector is then time in-variant. However, as the process is never entirely stationary, the Yule-Walker equation can only be true for *assumed* stationary processes, which is achieved by determining the process samples outside the framing window to be equal to zero (set of input values are segmented). In general, the selection of framing windows would also determine the estimation accuracy.

The autocorrelation function preserves information regarding the signal harmonics, formant amplitudes and periodicities, whilst ignoring phase. Applications of this process are commonly used for pitch detection, voiced and unvoiced speech determination, and most importantly linear prediction.

2.6.3 Robust Spectral Analysis

Autocorrelation method explained in the previous section is the most popular method of LP analysis. This method uses an all-pole (AR) model for estimating the power spectrum of a signal. For a signal that is affected by noise, the AR model would not be the correct model. The noise-corrupted signal would follow a pole-zero (ARMA) model [55]. The ARMA model of LP analysis (proposed by Atal and Schroeder in [56]) and the *denoising* of speech using such models have been researched in [57]. However the perceptual differences between the pole-zero models and the high-order all-pole models are not significant. For estimating the spectral envelope of such signals, either an ARMA model for the signal is assumed or the signal should be cleaned prior to applying the autocorrelation method.

There have been a number of different approaches in designing a robust spectral analysis method or improving the LP analysis. Examples of such research include the estimation of noise using past power spectral values [58], the application of a maximum likelihood estimation algorithm for spectrum estimation [59], a robust LP analysis design that represents glottal source waves using the hidden Markov model for application in speech distorted by noise [60], and the SEEVOC method, which ignores the spectral peaks affected by noise. The SEEVOC method, which is a method specifically designed for application in noisy environments, is discussed in further detail in the following section.

SEEVOC Method for LP Analysis

The Spectral Envelope Estimation Vocoder (SEEVOC) method is a technique that has been proposed to improve the performance of the conventional LP analysis method. Developed in the early 1980's, this method uses only the parts of the FFT-computed spectrum that are least affected by noise. Thus it tries to clean the spectrum of speech by ignoring the spectral portions more affected by noise. Its analysis deploys a methodology that ignores the low-level spectral peaks that may be a result of noise or sidelobe effects [61].

This method seems to perform well on speech signals having low fundamental frequency. However, for signals having high fundamental frequency, it does not perform well [62]. Accuracy of its spectral envelope estimate also depends heavily on *a priori* knowledge of the average signal pitch (for non-periodic waveforms),

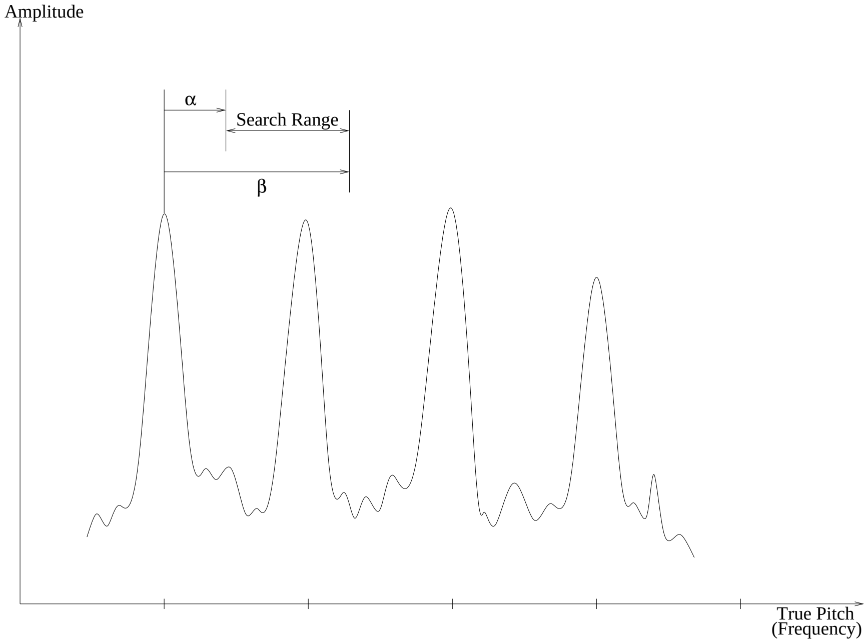

Figure 2.9: Methodology of the search process in SEEVOC.

which is a complication in real-world applications.

The SEEVOC method focuses on attaining the speaker pitch using an adaptive search method on the spectral peaks. The successful allocation of these spectral peaks and its associated amplitudes is what determines the robustness of this method. The frequency response of the vocal tract filter is sampled by the harmonics of the periodic impulse sequence. Spectral peaks are searched inside a certain interval [α, β] of the Coarse Pitch (CP), which is a small frame adaptively set. The SEEVOC spectral peak search algorithm is set for a finitely set frame, and done in a way that the true pitch of the spectrum can be obtained or best predicted.

Determination of search range to locate each spectral peak W_n would be dependent on the location of the previous peak W_{n-1}. Using the previous peak as an origin point of the search, the search procedure would cover the range [W_{n-1}+α, W_{n-1}+β]. For a common setting of $\alpha = \frac{1}{2}$CP and $\beta = \frac{3}{2}$CP, the search range would then be limited to CP frequency points. If no distinct peak can be located within the set range, the search would be continued using W_{n-1}+CP as the new origin point. Figure 2.9 illustrates the basic operation of this search.

The spectral peak allocation algorithm is performed together with an interpolation sequence to develop the desired spectral envelope function. A basic application of the SEEVOC method normally employs linear interpolation upon the spectral peaks, which will be the choice for further simulations in LP analysis. Another option is to perform linear interpolation upon the log spectral domain, however no significant improvement could be observed or has been reported. A third order spline interpolation, as proposed in [63], produces a slightly better quality spectral envelope estimate at the expense of the computational cost.

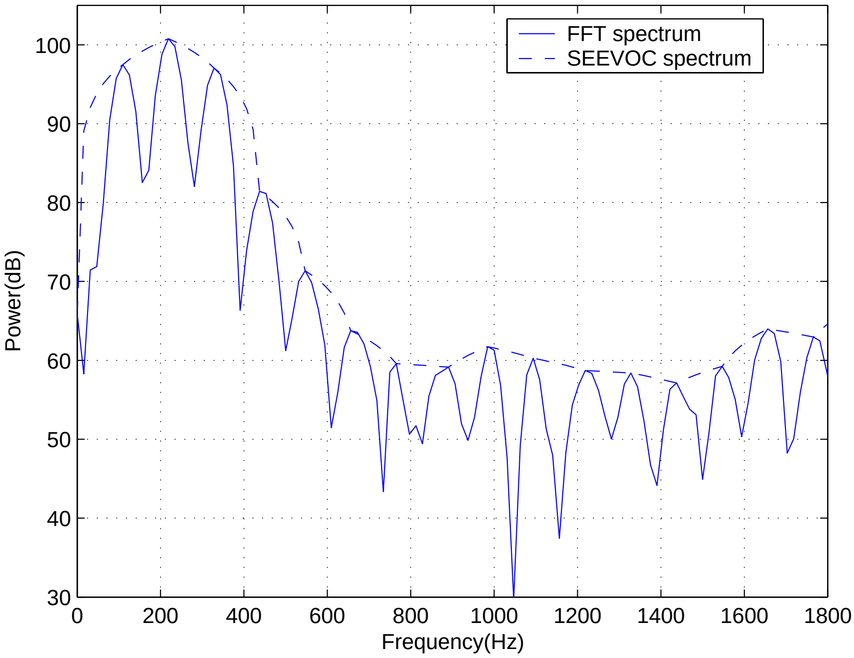

Figure 2.10: SEEVOC power spectrum after allocation of peaks.

Figure 2.10 shows the spectral envelope resulted from the SEEVOC algorithm with linear interpolation. Figure 2.11 compares the SEEVOC spectrum after LP analysis (SEEVOC-LP) with the autocorrelation method of LP analysis (AM-LP). These

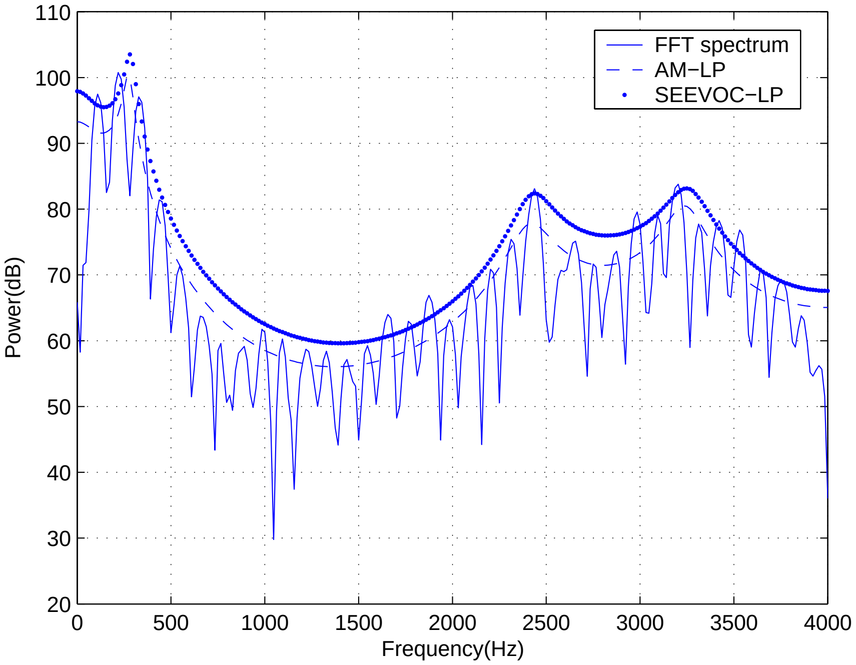

Figure 2.11: SEEVOC spectral envelope after linear prediction.

figures are presented to aid the understanding of the operation of the algorithm.

These results were computed for a speech segment [e] over a 30 ms time frame. Coarse Pitch was chosen to be 10 samples with search settings $[\frac{CP}{5}, CP]$ and LP order of 10. If the search range, set by α and β, is too large, then only the largest peak out of the multiple spectral peaks inside the search area is considered. On the other hand, a narrow search area would cause the search algorithm to allocate any peak (does not necessarily constitute a harmonic peak) to its SEEVOC spectrum.

The importance in the selection of CP can be seen in Figure 2.12. The true pitch (TP) of the signal is calculated by averaging the separation between all the harmonic spectral peaks. It can be observed that minimum distortion was achieved when CP approaches closer to TP. Spectral distortion (SD) measurement is performed on the spectral peaks of the FFT-computed power spectrum and the SEEVOC spectral envelope after LP analysis. TP was noted to be approximately 6.8

frequency samples (corresponds to approximately 106 Hz for FFT length of 512 frequency samples), and therefore its SD is noted to be at its minimum for CP in the range of 5 to 10 frequency samples (consistent with TP). It is therefore crucial to ensure that the search range is not larger than twice the length of TP.

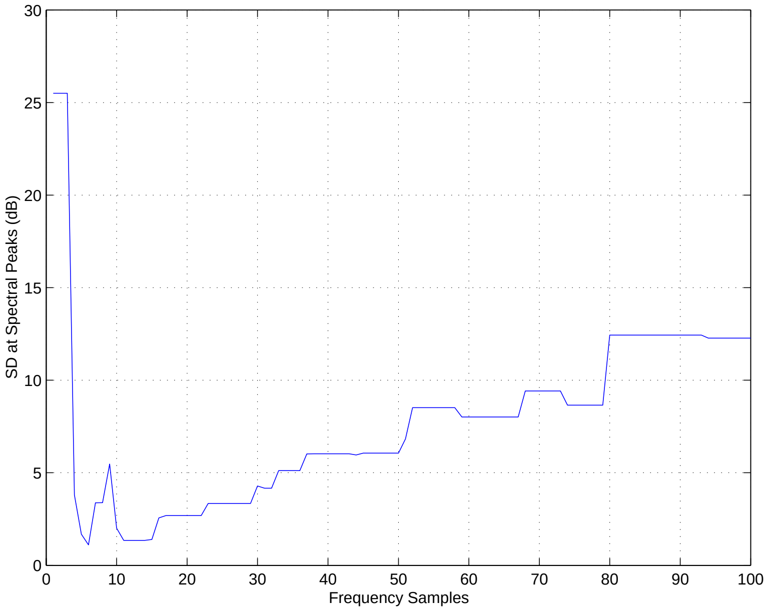

Figure 2.12: The effect of CP selection with TP=6.8 frequency samples.

From the above explanation, it can be seen that this method has disadvantages in needing knowledge of the true pitch of the signal and its dependence on peak allocation. It has been documented to work well on signals with low fundamental frequency but not on signals with high fundamental frequency. The spectral peaks in signals with high fundamental frequency, determined by its harmonic pitch, are located in close proximity to each other. The choice of CP would need to be as close as possible to the signal's TP in order to achieve an effective allocation of those spectral peaks.

Further discussion regarding the performance of the SEEVOC method in comparison to the autocorrelation method (AM) and the robust LP analysis methods will be covered in the later chapters.

2.6.4 Determination of the LP Parameters

The LP analysis produces a spectrum represented by a time-varying all-pole digital filter, where its transfer function $H(z)$ is represented as follows,

$$H(z) = \frac{G}{1 - \sum_{k=1}^{p} a_k z^{-k}} \tag{2.27}$$

where G represents the gain parameter (determined by its prediction error), p the LP order and a_k the LP parameters.

It is necessary to limit the LP analysis to short-time blocks of filtered signal sequences. The order of the LP analysis is normally dictated by the signal's sampling frequency. For a speech signal with sampling frequency of 8 kHz, with each 1 kHz proposed 1 pole and 2 poles allocated for the beginning and end of each analysis frame, a 10^{th} order LP is normally used (a_1, a_2, $\ldots$, a_{10}), with $a_0 = 1.0$ (representing the weighting for the current sample as can be seen in (2.27)).

The approach taken to solve the Yule-Walker equation (2.25) is the *Levinson-Durbin*[7] algorithm used for the autocorrelation method. This AR method of spectral estimation has also been referred to as the maximum entropy spectral estimation method [Burg, 1975]. This algorithm exploits the Toeplitz property of the autocorrelation matrix R_{xx} (please refer to previous section). From (2.24), the autocorrelation function can be calculated for input signal x_i with frame size M.

$$R_{xx}(k) = \frac{1}{M-k} \sum_{i=1}^{M-k} x(i)x(i+k) \tag{2.28}$$

[7] *This recursion was originally developed by Levinson in 1947, to solve the common matrix problem $Ax = b$, where A is Toeplitz, symmetric and positive definite and b arbitrary. Durbin in 1960 produced an efficient manner in which to solve the problem, developing the Levinson method to the recursion that is commonly used now.*

Initially,

$$a_1(1) = \frac{-R_{xx}(1)}{R_{xx}(0)} \tag{2.29}$$

and

$$P_1 = R_{xx}(0) \times (1 - |a_1(1)|^2) \tag{2.30}$$

Thus for the next steps, for $m = 2, 3, \ldots, N$,

$$a_m(m) = \frac{-\left[R_{xx}(m) + \sum_{i=2}^{N}(a_{m-1}(i) \times R_{xx}(m-i))\right]}{P_{m-1}} \tag{2.31}$$

$$P_m = P_{m-1} \times (1 - |a_m(m)|^2) \tag{2.32}$$

where N is the LP order, and solving for the LP parameters (for i = 1, 2, ..., $m - 1$),

$$a_m(i) = a_{m-1}(i) + a_m(m) \times a_{m-1}(m-i) \tag{2.33}$$

$$a(i) = a_N(i) \tag{2.34}$$

LP analysis treats problems of adaptive linear systems and explains it to sets of measurements or observations. Such observations in this case are used to analyse and estimate the power spectrum envelope of a given data set. This analysis would remove neighbouring sample correlations present in the signal, which ultimately can be used to estimate the spectral envelope in the frequency spectrum. However, there are still effective variations in the signal spectrum.

The LP parameters developed from the linear prediction theory perform well as a spectral analysis tool. The coefficients obtained will vary slowly as time progresses, following the behaviour of speech itself.

2.7 Code Excited Linear Prediction Coder

2.7.1 Background Theory

In the past two decades since its introduction, countless number of publications has been released with the aim to improve the basic design of this coder. However, as the main area of this research is to achieve optimum performance regardless of the CELP design, hence the basic design of the coder is chosen to complete the desired simulations. Research into the improvements proposed for the CELP coder is done purely to gain more understanding of the subject.

From its early development, the CELP coder was designed to meet requirements needed for compression of signals below 8 kHz. The original development of CELP proceeded via the application of VQ in solving the problems of coding the excitation signals from multipulse LPC. This development has lead to less audible distortion along with bandwidth requirement for LPC design [64].

Due to the effects of windowing distortion, the window size used for obtaining the autocorrelation coefficients in LP analysis must include at least two pitch periods for accurate spectral estimates [52]. Short sequences of the LP residual signal are most commonly coded via VQ (please refer to Section 2.4.2). The main diagram for the CELP coder is set out in Figure 2.13.

The operational steps of the CELP coder can be explained as follows:

1. *LP analysis.* Partitioning of the original signal into short-time frames is done prior to analysis (normally set to 20 ms). An LP analysis is then performed on these frames to give a set of LP parameters for each frame to use in the short term predictor (STP) to estimate the spectral envelope of the data.
2. *Memoryless STP.* The STP memory for initial conditions is removed to provide a memoryless STP for subsequent analysis. The filter is defined as

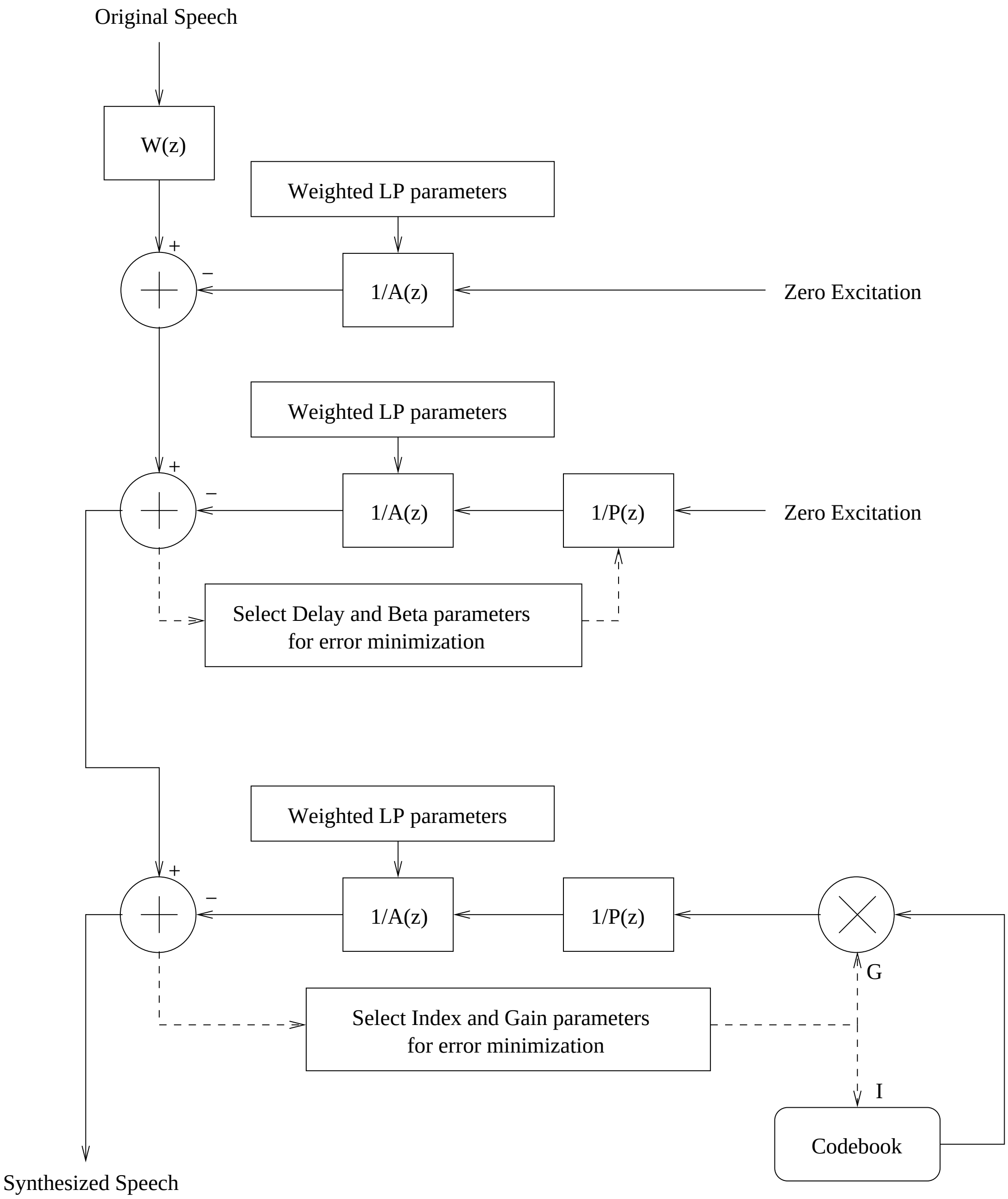

Figure 2.13: Block diagram of the CELP coder.

follows,

$$A_w(z) = \frac{1}{(1 + \sum_{k=1}^{N} a_k z^{-k})} \tag{2.35}$$

3. *Long Term Prediction (LTP) analysis.* This analysis is done on sub-frame lengths of 5 ms (set to be a quarter length of each frame). LTP analysis is performed on the residual or original signal computed using an inverse filter with the derived LP parameters, depending on the method used. This analysis computes the delay (D) and scaling coefficients (β) parameters and also introduces voice periodicity to the signal.

4. *Excitation determination.* The excitation can be determined from the two synthesis parameters found (codebook index and its respective gain). It is selected from a user-defined codebook, generated from white Gaussian sequences. The sequence that provides minimum error between the filtered Gaussian sequence (synthesised speech) and the original signal is then selected. Thus the selected codebook sequence produces the minimum squared objective error with its corresponding scaling factors.

5. *Reconstruction of synthesised signal.* In the synthesiser, the initial conditions of the filters are restored and the synthesised signal is computed by filtering the CELP information through the filters without any perceptual weighting.

For signal compression application of the robust LP analysis methods (Chapter 5), the CELP coder detailed in this section is used for the purpose of simulating its improvement in terms of quantisation distortion.

2.7.2 Quantisation of Pitch Parameters

Pitch Prediction in LTP Analysis

After LP analysis a long-term correlation still exists. In order to remove the periodic structure of the signal, a *pitch prediction* process is needed. The LTP analysis

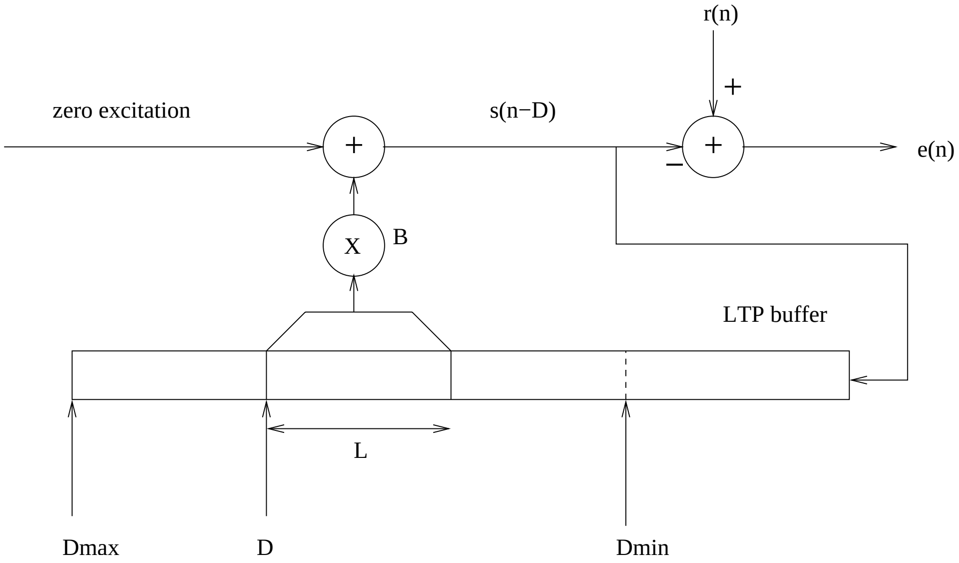

Figure 2.14: Block diagram of the long term prediction analysis.

exploits the correlation between the samples that are of a pitch period away.

Periodicity of the LP residual can be determined through the application of a pitch filter, which is dependent on its pitch gain coefficient (approaches unity gain for *exact* periodic signals) and lag coefficient (indicating its pitch periodicity) [65]. For a 7-bit CELP encoder applied on speech sampled at 8 kHz, there would be a possibility of 128 sample delays; hence an update rate of 5 ms will translate to a delay range of 40 to 167 samples. Determination of the pitch filter parameters is normally done using a closed-loop analysis-by-synthesis method, however an open-loop method is still used to determine its initial coefficient settings.

The fundamental formula used to perform the LTP analysis can be seen in (2.36). This analysis is aimed to model the long-term correlation in the signal residual after LP analysis by subtracting the LTP parameters. Once the delay D (which approximates the pitch period or periodicity of the signal) and scaling coefficients β (which determines the level of periodicity, with 1 corresponding to a 'perfectly' periodic signal) parameters have been obtained, they would spectrally flatten the signal spectrum. The pitch synthesis filter, $P(z)$, is defined as follows,

$$P(z) = \frac{1}{(1 - \beta z^{-D})} \tag{2.36}$$

Using an open-loop method and assuming a single-tap filter, β can be obtained from the following calculations,

$$\beta = \frac{\sum_{n=0}^{L-1} r(n)s(n-D)}{\sum_{n=0}^{L-1} s^2(n-D)} \tag{2.37}$$

where L represents the frame size (please refer to Figure 2.14). The output signal from the LTP phase would then be

$$e(n) = r(n) - \beta s(n-D) \tag{2.38}$$

where $r(n)$ is the synthesised signal at this point (after pre-emphasis filtering) and $s(n)$ is the original predicted signal.

Quantisation

No consistent correlation between pitch delay D and pitch gain β exists, hence the parameters are quantised individually [66]. The typical update rate of 5 ms results in 200 pitch coefficient updates for every second of speech. The pitch delay is encoded with a 7-bit quantiser (128 possible pitch delays), which covers the range of 40 to 167 samples for an update rate of 5 ms.

The pitch gain is treated as vectors, which may be quantised using 1-2 bits to represent the fractions for short delays. However it is normally quantised with a 3-bit vector quantiser using the LBG algorithm. This translates to a bit-rate of 2 kb/s for the pitch information. The VQ codebook is computed using a large collection of gain coefficients taken from CELP coder simulations performed previously. Originally the pitch delay is determined from the un-quantised gain factors, then using that delay value, the optimum quantised pitch gain coefficients can be calculated by calculating its minimum distortion.

2.7.3 Quantisation of Gain Parameters

Excitation Codebook

Stochastic codebooks are generally used to model the variations of voiced excitation. Therefore only the index representation of the codebook is needed to be transmitted, where both transmit and receive ends having identical codebook settings.

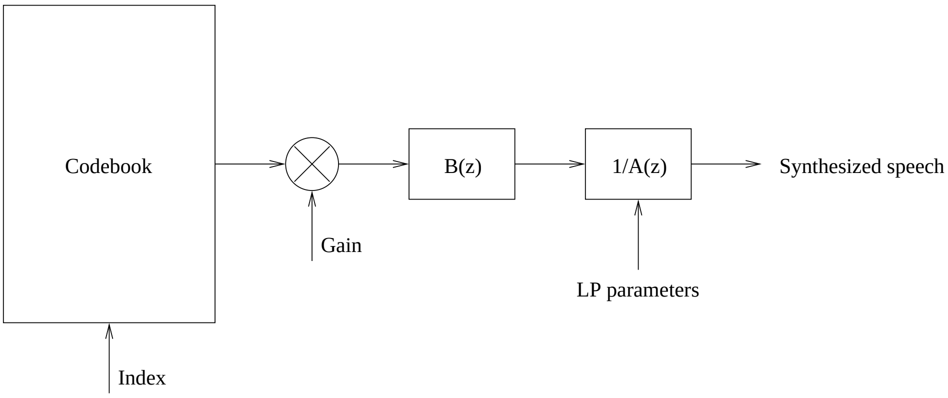

Figure 2.15: Basic block diagram of the codebook computation procedure.

The computation process, which is used for determining the excitation parameters, uses a codebook formed by random Gaussian noise. This is caused by the fact that signals after STP and LTP analysis would form into samples having behaviour similar to Gaussian signals. An obvious obstacle in this choice would be the size of the respective codebook. A 7-bit codebook would need $2^7 \times L_v$ of storage data space, where L_v represents the codebook vector size. To overcome this problem an *overlapping codebook* may be used, such that the codebook vectors would shift one or two data-points from one consecutive vector to another.

Selection of codebook vector is computed by searching the optimum gain as follows,

$$Gain(n) = \frac{\sum_{n=0}^{L_v-1} r(n)c(n)}{\sum_{n=0}^{L_v-1} c^2(n)} \tag{2.39}$$

for

$$\frac{\left[\sum_{n=0}^{L_v-1} r(n)c(n)\right]^2}{\sum_{n=0}^{L_v-1} c^2(n)} = P_{optimum} \tag{2.40}$$

where *c(n)* represents the codebook vector.

As can be seen in Figure 2.15, the process of manipulating the source data and filtering the excitation involves the use of two filters. The first filter is the AR filter, which is an all-pole filter, hence the linear filter $H(z) = \frac{1}{A(z)}$. The filter's input-output relationship can then be defined as follows,

$$x(n) + \sum_{k=1}^{p} a_k x(n-k) = y(n) \tag{2.41}$$

for p filter coefficients.

The other filter is the *Moving Average* (MA) filter, where the linear filter $H(z) = B(z)$ is an all-zero filter and

$$x(n) = \sum_{k=1}^{p} b_k y(n-k) \tag{2.42}$$

It has been observed that manipulation of codebook dimension does not significantly improve its performance improvement rate. Improvements are regarded linear as the dimensionality of each codebook increases. CELP simulation to help determine the dimension of the codebook can be seen in Figure 2.16. Codebook dimension was varied from 1 to 10 bits using codebook vector sizes of 5 ms. The input signals used are 8 kHz speech samples generated from 3 male and 2 female speakers reading identical sentences (each spanning approximately 3 seconds). Please refer to Section 3.5 for description of database used for this simulation. The following settings were used: FFT size 512 frequency samples, pitch delay 128 samples, LP analysis was performed with an order of 10 on a 20 ms frame. A 7-bit random Gaussian codebook is chosen to be a reasonable option for further simulations performed in Chapter 5.

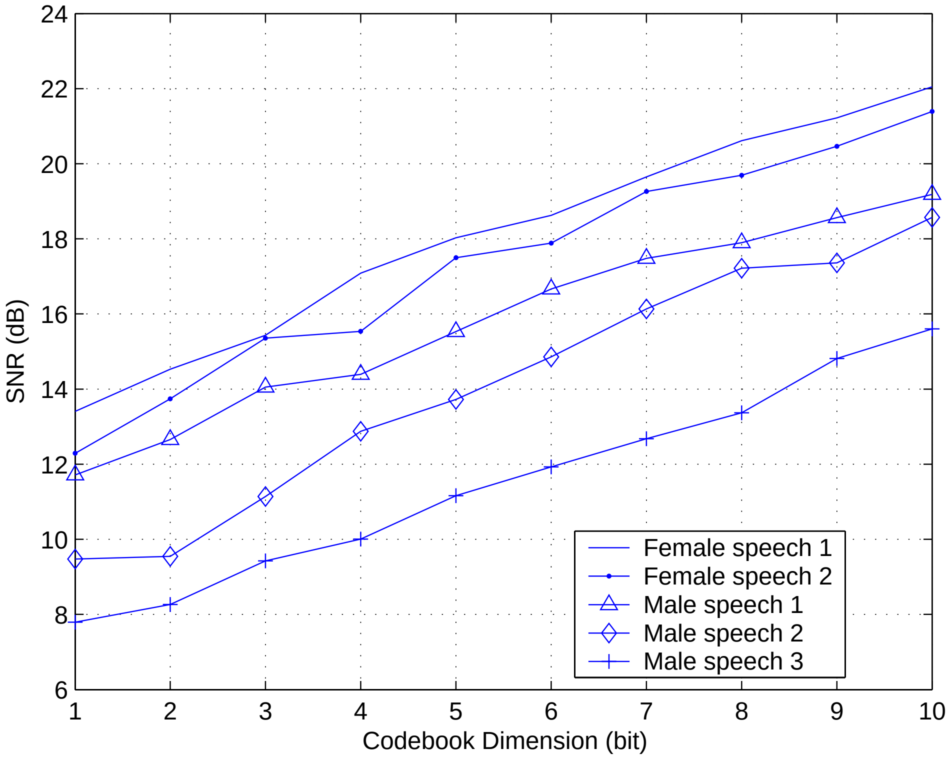

Figure 2.16: SNR performance for different codebook dimension.

Quantisation

Codebook gain parameters are encoded using a non-uniform quantiser using the LBG algorithm to compute its quantiser codebook [67]. Allocation of 3-4 bits per update for the excitation codebook gain parameter is combined with the assignment of a 7-bit codebook. This results in a bit-rate of 2 to 2.2 kb/s for the codebook information.

2.7.4 Quantisation of LP Parameters

Introduction

Quantisation of the LP parameters, and its transformations, utilises the LBG algorithm, as discussed in Section 2.4.2. Quantisation of residual signals using PCM and VQ methods has been widely applied for transmission purposes. However the LP parameters that are contained as side information in transmission still poses as a problem. In LP analysis, the quality and intelligibility of a coded speech relies heavily on the accuracy of the estimated envelope of the power spectrum, which in turn determine its LP parameters.

It is known that LP parameters are crucial in spectral analysis as it describes perceptually important spectral peaks in a frequency plane [31], [68]. An accurate spectral envelope can only be generated after quantisation with minimum degradation of the resolution of the spectral parameters. A considerable amount of transmission bit is normally needed in order to transmit the LP parameters. It should be pointed out that for an 8 kHz speech signal with an LP order of 10, each frame set of LP parameters would normally use between 30 to 60 quantisation bits, depending on the method of quantisation. For each transmission frame, that is a considerable amount of information even after quantisation. This has brought forth continuous study into the area of quantisation for transformed LP parameters. The basic theory involved with each transformation method will be discussed, with applications using scalar quantisers simulated. Applications of the transformation using VQ will be discussed in Chapter 4.

Quantisation Performance Criteria

Observations are made based on the performances of the LP parameters and its transformations applied with both uniform and non-uniform SQ, described later in this section, and VQ, described in Chapter 4. Simulations performed will show that, as expected, non-uniform SQ provides less quantisation distortion than the uniform SQ. Both non-uniform SQ and VQ employs the LBG algorithm in its application for LP parameter quantisation. The performance of quantisation on the LP parameters is computed using the *spectral distortion* (SD) measure (2.52) later discussed in Section 2.8.1.

Minimisation of distortion resulted from the quantisation of LP parameters is crucial in spectral analysis. The all-pole filter design has to remain stable after a quantisation process. Quantisation of LP parameters often results in small quantisation error that in turn can produce large errors relating to its LPC spectrum due to the dependency among the parameters. This might affect the stability of its resultant all-pole filter.

To avoid the prospect of a significant quantisation distortion, a large number of bits are normally used for the quantisation process, which is not an efficient solution with respect to its bit-rate. This has developed the necessity of transforming the LP parameters to separate representations that are less sensitive to the quantisation distortion and hence ensures the stability of $H(z)$.

In order to measure the performance of the quantisation process, SD is observed in two separate classifications; that is the average SD for the entire data and the percentage of outlier frames. A frame is considered to be an outlier frame if its SD $\geq$ 2 dB. Outlier frames are divided into; a) the SD ranges between 2-4 dB, and b) SD $>$ 4 dB. A desired performance for the quantisation of LP parameters is reached when its *spectral transparency* has been fulfilled, which is defined when:

- average SD $\simeq$ 1 dB,
- no outlier frame $>$ 4 dB,
- number of frames with SD between 2-4 dB is less than 2 % of the number of total frames.

It should be noted that quantisation is applied for the LP parameters defined using the autocorrelation method. The covariance method of LP analysis does not always produce a stable synthesis filter $H(z)$. Improvements upon this method have been successful in stabilising $H(z)$ at the cost of degrading the accuracy of the spectral envelope [64].

Database

Simulations performed in this chapter use the *TIMIT* database. The speech is resampled at 8 kHz with resolution of 16 bits per sample. A 10^{th} order LP analysis is performed using the autocorrelation method with analysis applied on 20 ms frames windowed with the hamming window. A 10 Hz bandwidth widening is applied (please refer to Section 2.6.1) to compensate the sharp spectral peaks. Just less than 236 minutes of speech (462 speaker for training, each reading 10 separate sentences) is used for training, while 28 minutes of speech (168 speakers separate from the training speakers, each reading 3 sentences) is used for testing. In total there are 707438 LP vectors used for training and 85353 LP vectors for testing (ratio approximately 8:1).

Transformation Methods

Quantisation of the LP parameters without any transformation can be observed in Table 2.1. It can be observed that using the autocorrelation method (AM), an average SD of approximately 1 dB can be reached with 80 quantisation bits, however spectral transparency can only be achieved with more bits. The extremely high bitrate highlights the obstacle initially mentioned regarding the need to remodel the LP parameters to different transformations that are more robust for quantisation.

Several transformations have been introduced in order to accommodate the spectral sensitivity of the LP parameters. It is crucial that these transformations should be

Table 2.1: SD performance of mid-level uniform scalar quantiser on LP parameters using uniform bit allocation.

Number of bits	*Average SD (dB)*	*Outliers (%)*	
		2-4 dB	*>4 dB*
70	1.256	16.436	1.370
80	0.693	4.248	0.082

able to map between each LP parameter and its transformed coefficient without any loss of information. The representations that have been developed and are covered here are the reflection coefficients, arcsine reflection coefficients (ASRC), log-area ratio (LAR), and line spectral frequencies (LSF).

Reflection Coefficients

Reflection coefficients are transformation coefficients developed for quantisation purposes, originating from the Levinson-Durbin recursion. These coefficients are referred as so due to its relationship to the reflection coefficients of the acoustic tube models of the vocal tract [69]. This transformation (K_i as the i^{th} reflection coefficient), also known as the partial correlation (PARCOR) ladder form, is spectrally less sensitive to quantisation than LP parameters and the all-pole filter stability is ensured (coefficient range is always inside $-1 \leq K_i \leq 1$ during quantisation). PARCOR coefficients are transformed from the LP parameters (a_i^i) of order p as defined below,

$$\begin{array}{ll} K_i = a_i^i, & i = p, p-1, \ldots, 1 \\ a_j^{i-1} = \frac{a_j^i + K_i a_{i-j}^i}{1-K_i^2}, & 1 \leq j < i \end{array} \tag{2.43}$$

Alternately, the reverse process can be defined as follows,

$$\begin{array}{ll} a_i^i = K_i, & i = 1, 2, \ldots, p \\ a_j^i = a_j^{i-1} - K_i a_{i-j}^{i-1}, & 1 \leq j < i \end{array} \tag{2.44}$$

The PARCOR transformation method is known to have limitations with regards to its performance in low bit-rate systems. The quality of the quantised PARCOR coefficients depends heavily on the number of bits that can be allocated for quantisation, which is a significant number (around 40-60 bits per set of 10 coefficients). This has brought forth research into better transformation methods for LP parameters.

With uniformly spaced SQ following the mid-level uniform PCM method, the following simulations are performed using uniform bit allocation for individual parts[8].

Table 2.2: Performance of mid-level uniform SQ on PARCOR coefficients.

Number of bits	*Average SD (dB)*	*Outliers (%)*	
		2-4 dB	*>4 dB*
40	1.299	15.430	0.794
50	0.679	2.012	0.001

Table 2.3: Performance of non-uniform SQ on PARCOR coefficients.

Number of bits	*Average SD (dB)*	*Outliers (%)*	
		2-4 dB	*>4 dB*
40	1.028	7.506	1.286
50	0.517	1.653	0.098

Non-uniform SQ of the transformation parameters should achieve better performance because the parameters do not have a flat spectral behaviour. Specifically for PARCOR, the distribution of the parameters are more sensitive around 1 than 0, which leads to the understanding that non-uniform quantisation would be more beneficial for quantising the parameters. A study by Gray and Markel [70] also explains the effects of non-uniform bit allocation to achieve optimum performance. The determination of the optimum quantisation bit allocation is generally aimed to minimise its overall spectral distortion.

[8] *This bit allocation process assumes that all set of coefficients contain equal variance and is determined by at least 1 bit. As an example, for LP order of 4, a 6-bit quantiser would have an allocation of bits = [2 2 1 1], 7-bit quantiser would allocate its bits = [2 2 2 1] and so on.*

Arcsine Reflection Coefficients

This non-linear transformation was developed to expand the region near $|K_i|$=1 [51]. This is done to avoid the quantisation sensitivity for narrow bandwidth pole representations, which is the disadvantage of the reflection coefficients. The ASRC coefficients (J_i's) are defined from the reflection coefficients as

$$J_i = \arcsin(K_i), \qquad 1 \leq i \leq p \tag{2.45}$$

Hence,

$$K_i = \sin(J_i), \qquad 1 \leq i \leq p \tag{2.46}$$

Table 2.4: Performance of non-uniform SQ on ASRC coefficients.

Number of bits	*Average SD (dB)*	*Outliers (%)*	
		2-4 dB	*>4 dB*
40	0.931	4.621	0.554
42	0.700	2.438	0.285
44	0.591	1.295	0.096
46	0.528	0.842	0.049
48	0.481	0.729	0.049
50	0.457	0.714	0.047

Log-Area Ratio

This transformation was developed to exploit the same disadvantages associated with the reflection coefficients. The LAR coefficients (L_i's) are defined as follows,

$$L_i = log\tfrac{1+K_i}{1-K_i}, \qquad 1 \leq i \leq p \tag{2.47}$$

and in turn its reflection coefficients are defined below

$$K_i = \tfrac{1-e^{L_i}}{1+e^{L_i}}, \qquad 1 \leq i \leq p \tag{2.48}$$

Table 2.5: Performance of non-uniform SQ on LAR coefficients.

Number of bits	*Average SD (dB)*	*Outliers (%)*	
		2-4 dB	*>4 dB*
40	0.900	3.387	0.446
42	0.681	1.783	0.194
44	0.576	0.927	0.070
46	0.513	0.525	0.034
48	0.467	0.445	0.030
50	0.443	0.445	0.029

In [71] it has been observed that for speech signals with preemphasis filtering, LAR would perform better than PARCOR. However this is not always the case when applied on signals with different characteristics that may cause the LAR's to be unbounded (which can easily be fixed by artificial bounding of the values).

Line Spectral Frequencies

Also known as Line Spectrum Pairs (LSP's), it was first introduced by Itakura [72] and further developed in [73], [74], [75], [76]. This transformation method is a frequency domain representation that takes advantage of the properties of the human perception system. It exploits the modelling of speech via an all-pole filter. In [77] it has been observed that the LSF parameter locations are concentrated around the resonances of the LPC spectrum. If no resonances occur (e.g. silent frames) the LSF coefficients are distributed evenly throughout the frequency plane. The root of two polynomials, set out below, defines LSF.

$$\begin{aligned} P(z) &= A(z) + z^{-(p+1)}A(z^{-1}) \\ Q(z) &= A(z) - z^{-(p+1)}A(z^{-1}) \end{aligned} \tag{2.49}$$

where $A(z)$ represents the p^{th} order minimum phase polynomial, which can be translated back from the two $(p+1)^{th}$ order polynomials as follows,

$$A(z) = \frac{P(z) + Q(z)}{2} \tag{2.50}$$

Analysis filter $A(z)$ is generated directly from its LP parameters a_k,

$$A(z) = 1 - \sum_{k=1}^{p} a_k z^{-k} \tag{2.51}$$

In physical terms, the polynomials correspond to the lossless models of the vocal tract with glottis closed $[P(z)]$ and open $[Q(z)]$. The $P(z)$ and $Q(z)$ polynomials are defined by the property that all zeros are interlaced with each other and lie on the unit circle resulting in ascending LSF coefficients. Most importantly this ensures the stability of the all-pole filters. Thus far, this transformation method has been determined to be the most robust model for use in signal transmission. Improvements upon LSF on VQ by incorporating perception [78] or error shaping techniques [79] have also been researched for the purpose of lowering its quantisation bit-rate.

Table 2.6: Performance of non-uniform SQ on LSF coefficients.

Number of bits	*Average SD (dB)*	*Outliers (%)*	
		2-4 dB	*>4 dB*
36	1.185	6.691	1.201
37	1.054	5.153	0.444
38	0.947	3.774	0.212
39	0.863	2.576	0.098
40	0.804	1.807	0.087

Due to the interlacing between the LSF coefficients, another transformation method that manipulates that aspect has been developed called LSFD's (LSF Differences) [73]. This method however is highly sensitive toward channel errors thus producing large spectral distortion. Although a lot of research has been undertaken to improve the inferior quantisation performance of the LSFD's (including research of specifically design quantisers), it is still considered to be impractical and not computationally efficient. Further discussion regarding the quantisation performance of the LSF coefficients is set out in Chapter 4.

2.8 Performance Evaluation Criteria

2.8.1 Spectral Distortion Measure

In determining the quality of an estimated power spectral envelope, its *spectral distortion* (SD) is calculated over the power spectrum on a frequency plane as an objective measure.

$$SD = \sqrt{\frac{1}{N}\sum_{i=0}^{N-1}(10log_{10}\frac{P_i}{\hat{P}_i})^2} \tag{2.52}$$

where P_i and $\hat{P}_i$ are the true and estimated power spectrum respectively. As can be observed in the above equation, the reduction of the SD dictates the spectral envelope estimate's accuracy.

This distortion measure is normally performed upon the power spectrum computed from 20-30 ms length of speech, or equal to the frame length used for its LP analysis. This measure will be used to determine the accuracy and robustness of the proposed methods in Section 3.5.

2.8.2 Quantisation of the LP Parameters

Number of bits allocated for quantisation determines the efficiency of quantisation of the LP parameters. Generally the number of bits allocated for quantisation is determined only when the spectral transparency has been achieved. This will give an equal basis of comparison between the proposed methods and the more conventional LP analysis methods.

The quantisation process is separated into two sections, training and testing. Training the codebook involves the use of a pre-defined set of speech samples to generate training vectors consisting of its LP parameters. These training vectors are classified accordingly to form the quantisation codebook. Codebook design is dependable on the bit allocation settings and quantisation method. A stream of vectors, determined from a separate set of speech samples to that used for training, is used for testing. This set of test vectors is quantised according to the codebook computed from previous training procedure.

The train-test vector ratio of approximately 8:1 is determined to be sufficient for quantisation of LP parameters [68]. The evaluation of this process can be observed in Chapters 4 and 5.

2.8.3 Performance of the CELP Coder

In measuring the quality of a synthesised signal, as simulated in Chapter 5, the mathematical representation *signal-to-noise ratio* (SNR) is used.

$$SNR = 10log_{10}\left[\frac{\sum_{i=0}^{M-1} x(i)^2}{\sum_{i=0}^{M-1}(x(i)-y(i))^2}\right] \tag{2.53}$$

where x_i and y_i are the samples of the original and synthesised signals respectively, and M is the signal length. Signal quality would then naturally improve as the RMS difference is minimised.

In speech coding, SNR is a poor estimate of speech quality especially when a wide range of speech distortion is introduced. SNR is not specifically designed to model the subjective attributes of a speech signal; hence determining its speech quality would be unreliable.

In (2.53), the time domain error of a sequence of speech is weighted equally. This does not necessarily correspond well with the behaviour of speech where its energy varies with time. A quick solution to this problem is by applying the SNR calculation to speech on a frame-by-frame basis, thus weighting each short-time speech frame independently and then averaging the overall ratio [80]. This Segmental-SNR can be defined,

$$SNR_{Seg} = \sum_{j=0}^{N-1} 10log_{10} \left[\frac{\sum_{i=(n_j-M+1)}^{n_j} x(i)^2}{\sum_{i=(n_j-M+1)}^{n_j} (x(i) - y(i))^2} \right] \tag{2.54}$$

where the n_j's are the end points of each frame (frame length M) and N is number of frames.

SNR$_{Seg}$ allows an objective measure of speech quality by assigning equal weight to loud (large energy) and soft (small energy) portions of speech. Normally to avoid unnecessary large distortions in speech, *silence* needs to be identified and excluded (most likely located at the beginning and end of each sample sequence). It has also been widely understood that thresholds can be set at its extreme ends of the scale (e.g. ratios below 0 dB and above 35 dB are left out). This is done because the ratios outside the set threshold limits are regarded as not offering any contribution to the overall speech quality [81].

It should be pointed out here that the SNR, and SNR$_{Seg}$ respectively, is only a mathematical ratio in comparing the performance of a particular speech coder. Although it is a reasonably reliable mathematical representation of signal quality, it is still possible to have two synthesised speech samples with one sample having a worse SNR but better sound quality than the other. In the end, human subjects would still be needed to gather an subjective quality measurement, especially in producing synthesised speech.

This introduces the *Mean Opinion Score* (MOS), which is an average numerical opinion score for a set of untrained subjects. The MOS is the most commonly used subjective measure for determining signal quality. It uses human subjects to determine the quality of speech using a predetermined setting of 1 to 5, with 5 representing excellent quality.

Chapter 3

Robust LP Analysis Methods

3.1 Introduction

The performance of the autocorrelation method of LP analysis has always been limited when the speech signal is corrupted by noise. A new approach for LP analysis needs to be designed in order to overcome these limitations, which is explained in Section 1.2. A number of robust LP analysis methods have been proposed recently in [82] and [83]. These methods compute the LP parameters in two steps. In the first step, they manipulate the FFT-computed power spectrum with the aim of removing the effect of noise. In the second step, they apply the conventional autocorrelation method on the autocorrelation coefficients computed by taking the inverse FFT of the clean power spectrum.

Autocorrelation coefficients (R_{xx}) are calculated from the spectral envelope estimates, such that

$$R_{xx}(\tau) = \int_{-\infty}^{\infty} P_{xx}(f) e^{j2\pi f\tau} d\tau \tag{3.1}$$

where $P_{xx}(f)$ represents the distribution of the power spectrum as a function of frequency. The methodology used to calculate the LP parameters follows the LP analysis procedure described in Section 2.6.4.

The robust LP analysis methods are aimed at producing spectral envelopes, for use in speech coding applications, which are more robust than conventional methods when speech is introduced into noisy environments. The LP parameters should be able to model the analysis filter, such that the spectrum of the clean signal is maintained, whilst ignoring the spectral points affected by noise. In LPC applications, the synthesised signal is produced with less distortion as the spectral envelope calculated from the LP parameters is less sensitive to the effect of noise.

A robust LPC spectrum should not be estimated by simply locating the spectral peaks of a power spectrum and calculating its interpolated spectral envelope. This is not an accurate manner of calculating the spectral envelope since the introduction of noise would corrupt the lower-level peaks.

In order to minimise the effect of noise on the speech signal for LP analysis, the robust methods rely more on the harmonics peaks and ignore valleys between the harmonic peaks. Hence when noise is introduced, the estimated spectral envelope would maintain the general shape of the power spectrum, whilst not being affected by noise.

The three methods that are proposed are the moving average, moving maximum and average threshold method. Selecting the length of the analysis window of each method introduces a trade-off between the robustness and accuracy of the LPC spectrum. This will be investigated in the latter part of this chapter.

3.2 Moving Average Method

This method employs the moving average filter to smooth the FFT-computed power spectrum of the input signal. Using an M-size averaging window, the filter range

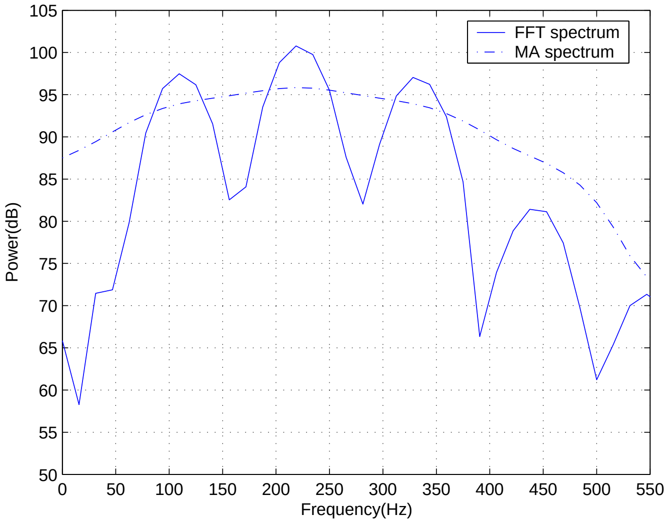

Figure 3.1: Spectral envelope after application of the moving average (MA) method for M = 25 frequency samples.

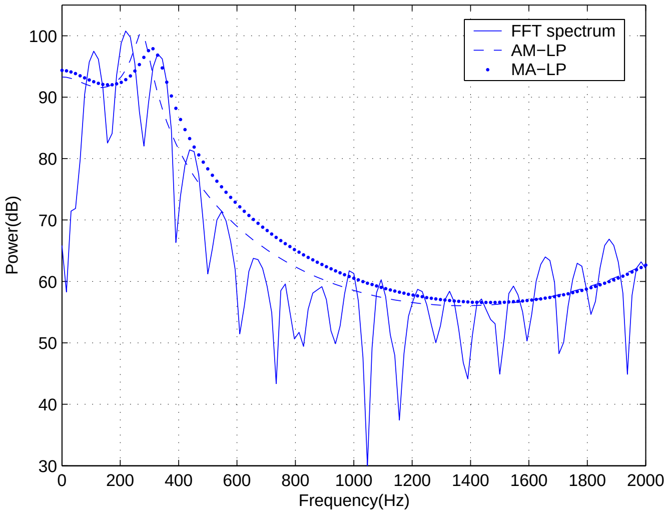

Figure 3.2: MA spectrum after LP analysis.

can be defined as

$$w(i) = \frac{N - |i|}{N^2} \tag{3.2}$$

for $-N \leq i \leq N$ and $N = \frac{M-1}{2}$.

Figure 3.1 shows the moving average spectrum after applying the smoothing filter on the FFT-computed spectrum. Figure 3.2 shows the moving average spectrum after LP analysis. The simulations were executed on a 30 ms speech segment [e] with LP order of 10. The moving average window was set to approximately 390 Hz, 25 samples over an FFT matrix length of 512. Further discussion regarding the performance of the proposed method mentioned here and on the two following sections will be covered in Section 3.5.

3.3 Moving Maximum Method

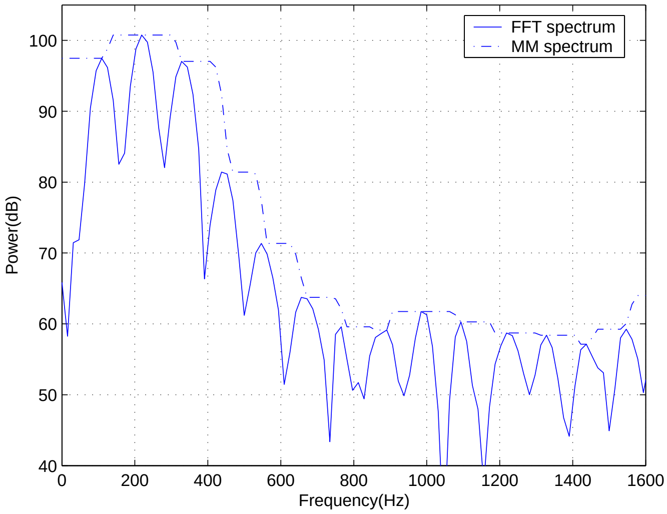

Figure 3.3: Spectral envelope after application of the moving maximum (MM) method for M = 11 frequency samples.

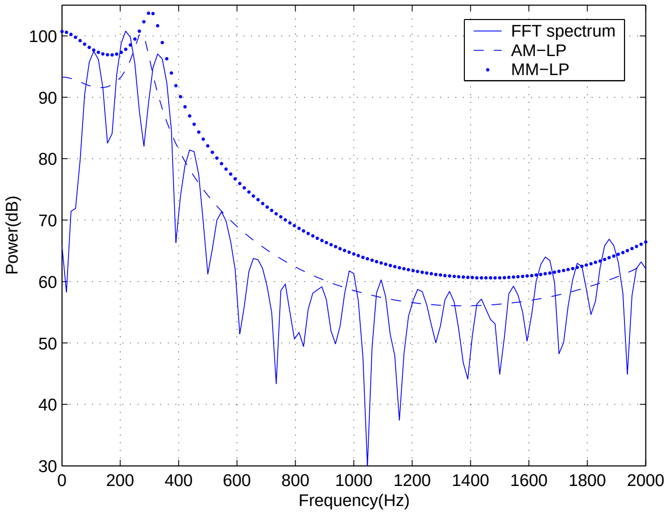

Figure 3.4: MM spectrum after LP analysis.

Moving maximum method searches for a maximum level from the FFT-computed power spectrum of the input signal over a defined range. The maximum point will then be used to represent a certain interval surrounding that frequency point.

Implementation of the moving maximum method is defined as follows:

- For each spectral point k in the frequency plane, the algorithm searches for a maximum value in the region of $[k - N,\, k + N]$.
- It then replaces the original value of that point with the resultant maximum value. The span of the moving maximum window would be (2N+1).

The result of this search method before and after LP analysis can be seen in Figures 3.3 and 3.4. Using the same settings described in Section 3.2, the moving maximum window is set to approximately 170 Hz over the power spectrum.

3.4 Average Threshold Method

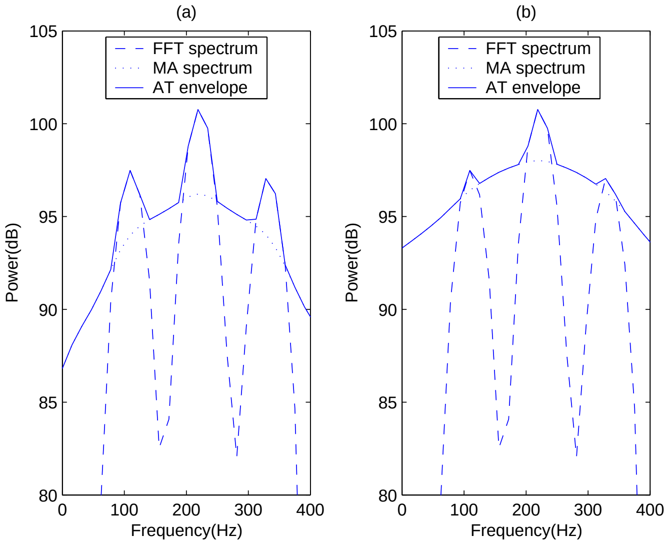

Figure 3.5: (a) Spectral envelope after first step of average threshold (AT) search method; (b) Spectral envelope after fourth AT repetition.

Average threshold method takes into account the benefit of both the moving average and moving maximum methods. It is based on a repetitive search of the FFT-computed power spectrum of its moving average spectrum, then taking its maximum in comparison to its FFT-computed power spectrum.

The methodology for the AT method is as follows:

- The moving average algorithm is applied on the FFT-computed power spectrum of the input signal.
- The resultant average spectrum is then combined with the original power spectrum. The larger value between the two spectral points at any given frequency locations is then used to form the new average threshold spectrum.

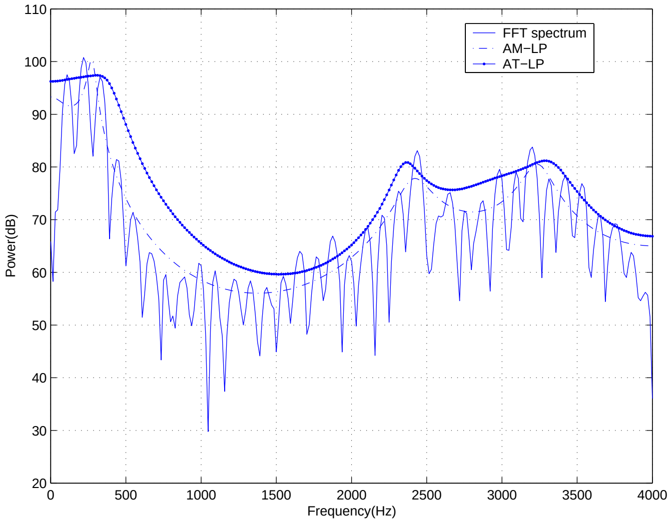

Figure 3.6: AT spectrum after LP analysis.

- The steps above are then repeated a certain number of times to achieve an optimum result.

Figures 3.5a and 3.5b show how this method is performed on the FFT-computed spectrum over a number of repetitions. The envelope spectrum after LP analysis can be seen in Figure 3.6 for 4 repetitions using moving average window of approximately 330 Hz. This method follows the objective of ignoring the low level peaks affected by noise to form a robust spectrum.

3.5 Robustness and Accuracy Analysis

3.5.1 Database

As applied in Section 2.7, the *TIMIT*[1] (Texas Instruments Massachusetts Institute of Technology) database is used for almost the entirety of this dissertation. The simulations use 462 train speakers and 168 test speakers (speakers are taken from 8 major dialects in the United States of America). The male-female speaker ratio is 70:30 across the 630 speakers. Each speaker reads 10 sentences, which is varied from a selection of 2342 different sentences, where each sentence read by any speaker is ensured not to sound identical to any of the other sentences. This database covers all the phonemes, with each phoneme appearing multiple times within different contexts.

This database, which was originally sampled at 16 kHz with 16-bit resolution, has been re-sampled at 8 kHz with identical resolution. Spectral envelope estimation is performed on the FFT-computed power spectrum with length of 512 frequency samples. A 10^{th} order LP analysis is performed on 20 ms analysis frames unless otherwise stated. It has been studied in [66] that the performance of the CELP coder degrades when the order of LP analysis is below 10. Bandwidth widening of 10 Hz is applied ($\gamma = 0.996$). The train-test vector ratio of approximately 8:1 is determined to be sufficient for quantisation of LP parameters in later chapters.

The noise samples from the *Aurora*[2] database are used in our experiments to simulate real-world noise conditions. The noise samples are varied for different SNR values (ranging from 35 dB, 30 dB, ..., -5 dB, -10 dB).

[1] *This database is a joint effort from various US sites under the sponsorship of the Defence Advanced Research Projects Agency - Information Science and Technology Office (DARPA-ISTO), released in October 1990.*

[2] *This database was created at the Ericsson Eurolab Germany in January 2000. It consists of background noises that are commonly encountered such as sounds from people dining in a restaurant, cars and pedestrians on a street, trains, and motor vehicles.*

3.5.2 Procedure

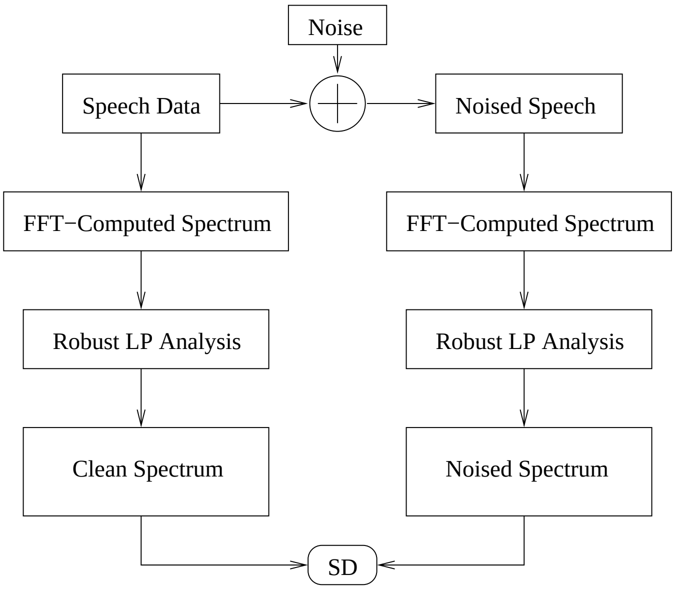

Figure 3.7: Methodology to simulate robustness.

In this chapter, the robustness and accuracy of the moving average (MA), moving maximum (MM) and average threshold (AT) methods are studied. The effect of varying the analysis window lengths is investigated for each method. The performance of the proposed methods will be compared to the conventional autocorrelation (AM) and SEEVOC methods of LP analysis.

The procedure to determine the robustness of the proposed methods uses the configuration set out in Figure 3.7. SD (2.52) is measured between the spectrum of the robust LP analysis method on clean speech and the spectrum of the robust method on speech signal affected by noise. As the level of noise distorting the speech is increased, the LPC spectrum resulting from the proposed method is expected to keep its general shape. Thus the spectral envelope generated by the proposed methods is expected to generally maintain its vigour even as the lower-level peaks are masked

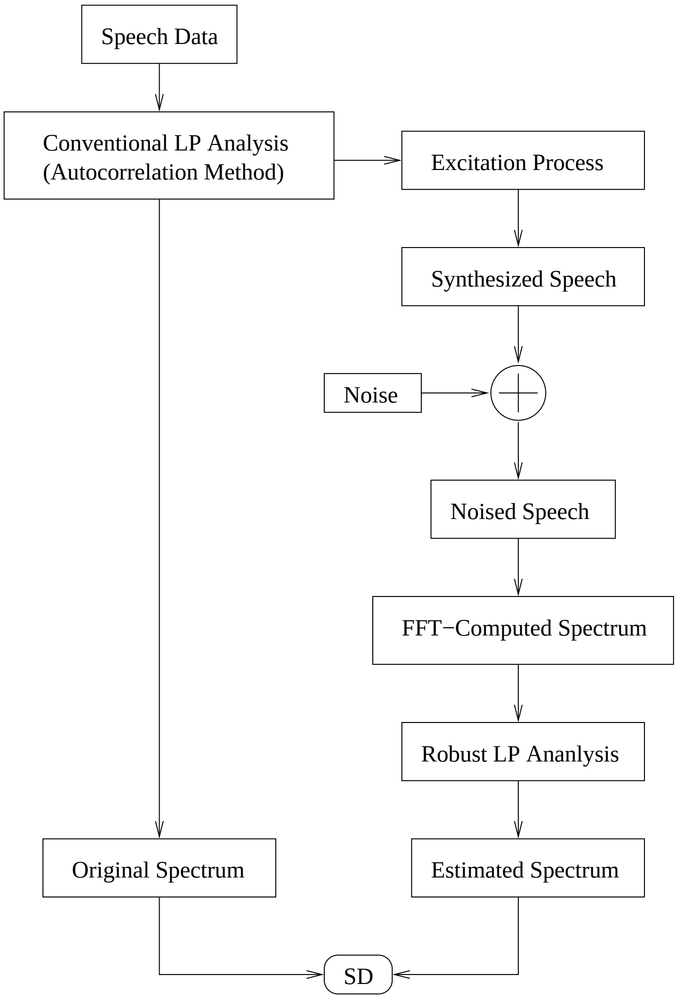

Figure 3.8: Methodology to simulate accuracy for the proposed methods.

by the power of noise. It should be noted that as the aim of this research is not to ensure complete isolation from noise, hence distortion of the spectral envelope is still observed as the effect of noise is increased.

The accuracy of the proposed methods in determining the *true* spectral envelope of a speech signal is simulated using the methodology displayed in Figure 3.8. In order to observe the performance of the methods in an objective manner, it is imperative to compare the methods on an equal plane. The following methodology is applied:

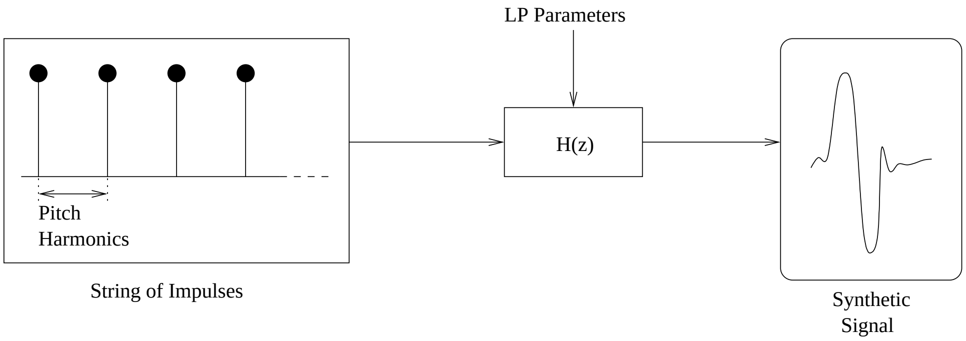

Figure 3.9: Excitation process to construct synthetic signal from LP parameters.

1. Using the conventional autocorrelation method for LP analysis, a set of LP parameters is generated from a speech sample.

2. A synthetic speech is generated from the LP parameters (please refer to Figure 3.9). This is achieved by applying an AR filter $H(z)$ on a string of impulses, which is defined as follows,

$$H(z) = \frac{1}{1 - \sum_{k=1}^{p} a_k z^{-k}} \tag{3.3}$$

   The filter tap of $H(z)$ is set by the LP parameters. The string of pulses is separated by a pre-defined pitch harmonic. As an example, for an 8 kHz signal, pulses pitched at 80 samples apart would give a frequency spacing of 100 Hz.

3. An estimated spectrum $\hat{P}$ is then calculated from the synthesised speech.

4. SD is computed by measuring the distortion between the original spectrum generated from the LP parameters in the first step and $\hat{P}$.

3.5.3 Results

The following simulations (as displayed in Figures 3.10 to 3.17) were simulated using 12 seconds of speech, spoken by three different individuals (1 female and 2 male speakers). Noise introduced to the speech was the *train* sample from the *Aurora* database, unless stated otherwise. The synthesised signal for the accuracy analysis was constructed using a pitch harmonic of 80 samples, which translates to $f_0 = 100$ Hz.

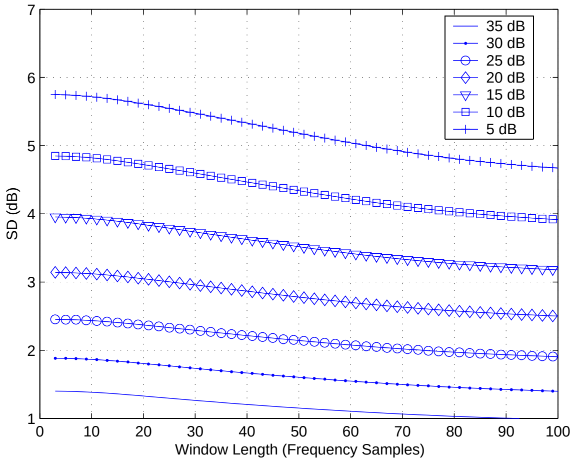

Figure 3.10: Robustness analysis of the MA spectral estimation method for speech signal affected by varying levels of noise.

Figure 3.10 shows robustness analysis for MA method. The noise level was varied for SNR of 35 dB to 5 dB. Improvement in robustness can be observed as the analysis window length is increased.

Figure 3.11 shows the accuracy analysis for the MA method of LP analysis. The size of the analysis window length of the MA method heavily affects the accuracy of the spectrum. It is concluded that the robustness performed by the MA method is compensated by the decrease in its accuracy.

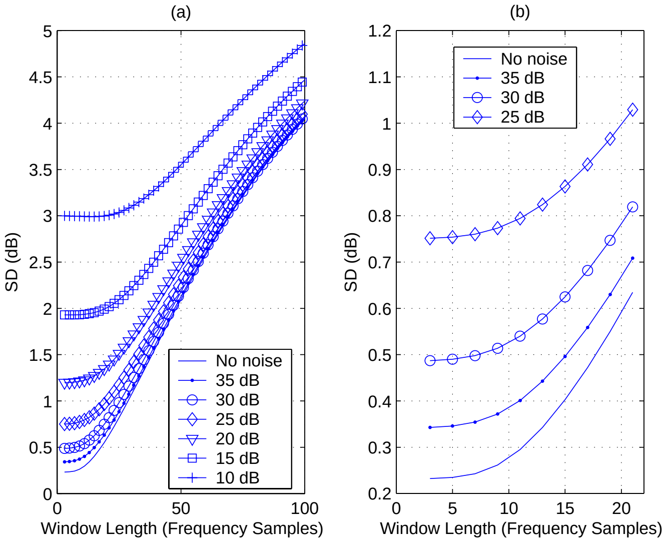

Figure 3.11: Accuracy analysis of the MA spectral estimation method, (a) for speech affected by noise; (b) closer look at signals with minimum noise introduction.

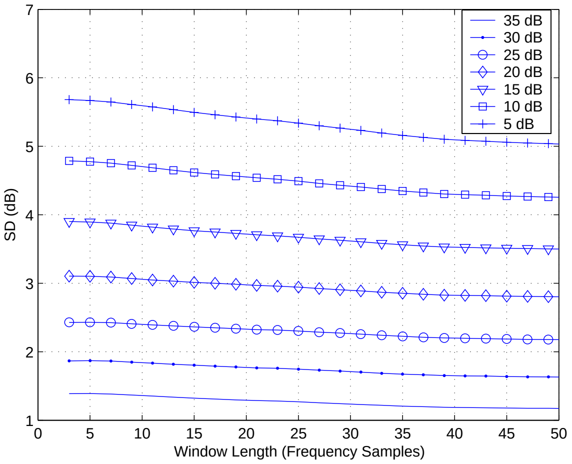

Figure 3.12: Robustness analysis of the MM spectral estimation method for speech signal affected by varying levels of noise.

The above case also holds true for the MM method of LP analysis. The robustness, shown in Figure 3.12, is compensated by the decrease in accuracy, shown in

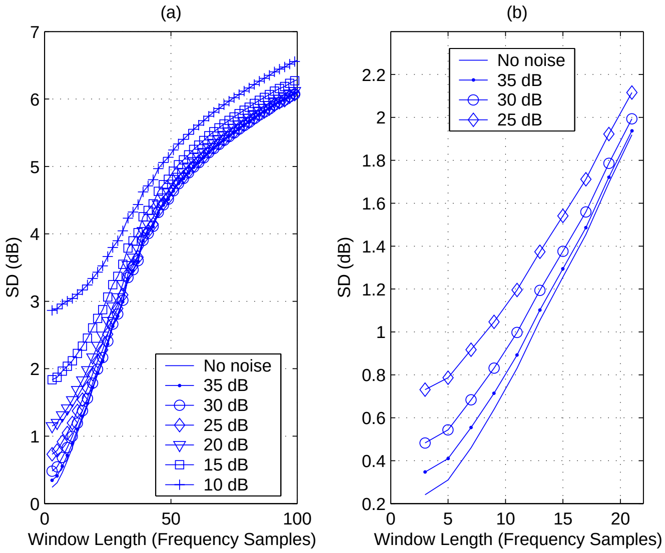

Figure 3.13: Accuracy analysis of the MM spectral estimation method, (a) for varying noise levels; (b) closer look at signals with minimum noise introduction.

Figure 3.13.

Simulations for the AT method uses the *restaurant* noise, which is a recorded sequence of background noise encountered when dining in a restaurant environment. For a fixed window length, the performance of the AT method can be seen for different number of repetitions. Robust simulation in Figure 3.14 shows an expected behaviour as the MA and MM methods. As the number of repetition increases, the accuracy decreases (as shown in Figure 3.15), however the rate of decline in SD is not as severe as the other two methods. This is consistent with its design methodology, where its power spectrum envelope follows the behaviour of the harmonic peaks more closely as the number of repetition increases.

Clearly there exists a trade-off between the accuracy and its robustness of the spectrum estimation methods. Figure 3.16 displays the trade-off between the two aspects for different number of repetitions (SNR = 25 dB). A narrow AT analysis window provides a less robust spectrum than a wide analysis window.

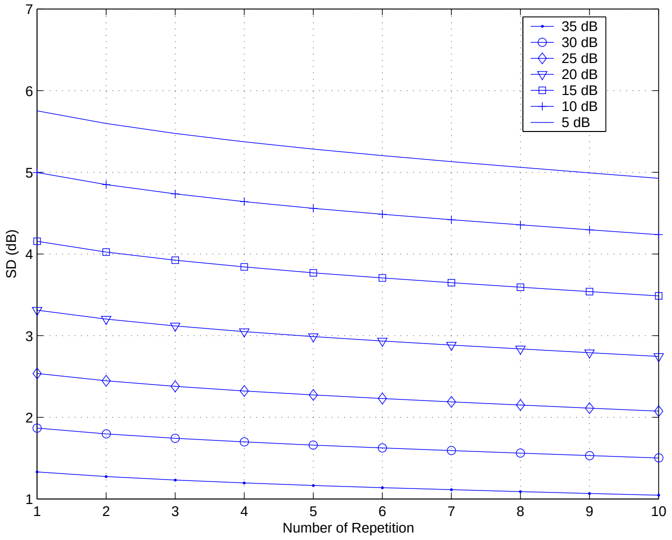

Figure 3.14: Robustness analysis of the AT spectral estimation method for speech signal affected by varying levels of noise (M=49 frequency samples).

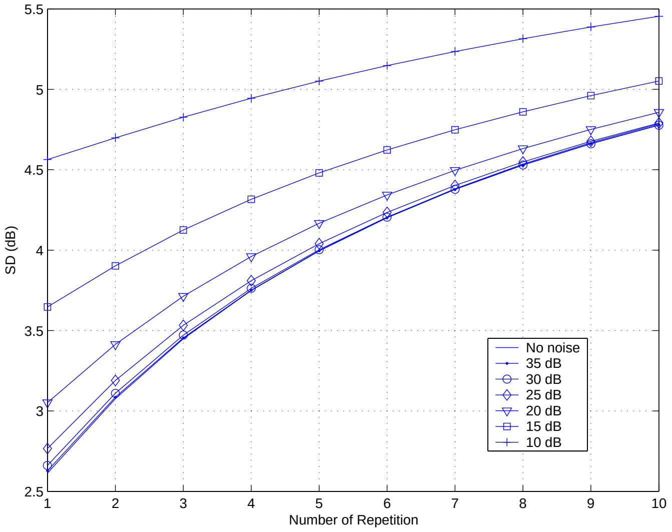

Figure 3.15: Accuracy analysis of the AT estimation method for speech signal affected by varying levels of noise (M=49 frequency samples).

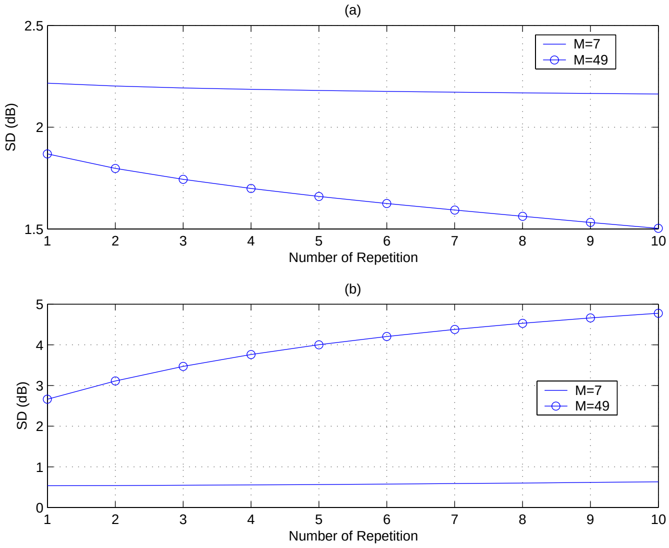

Figure 3.16: Comparison of different lengths of analysis window (M) on its (a) Robustness and (b) Accuracy.

Table 3.1: Relation between the level of noise and the SNR of the speech samples affected by noise for Figures 3.17 to 3.21.

Level of noise	*SNR (dB)*	*Level of noise*	*SNR (dB)*
0	∞	6	10
1	35	7	5
2	30	8	0
3	25	9	-5
4	20	10	-10
5	15		

The effect that the choice of number of repetition has on the performance of the AT method can be observed in Figures 3.17 and 3.18. These results were simulated on a 30 ms speech sample [*e*] introduced with Gaussian noise. Each line represents the performance of this method for repetitions of 1 to 10 times (as marked in the plots).

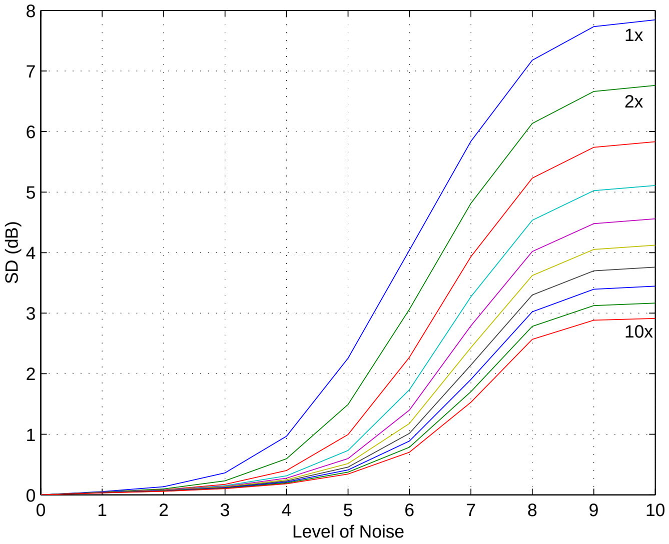

Figure 3.17: The effect of increasing AT repetitions on its robustness for different levels of noise (please refer to Table 3.1).

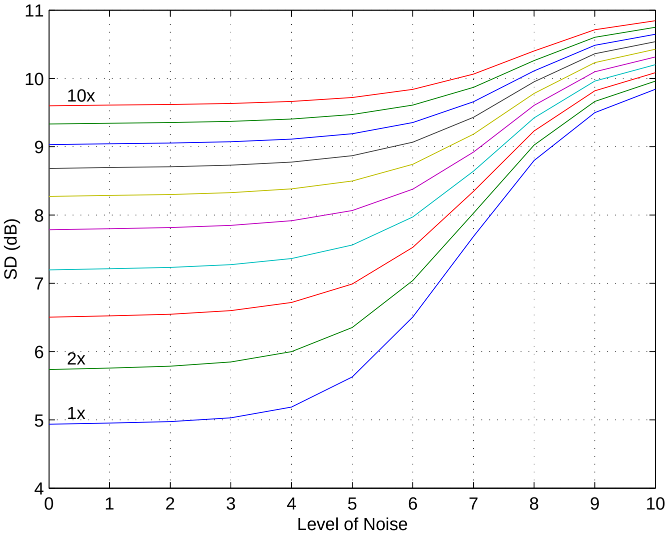

Figure 3.18: The effect of increasing AT repetitions on its accuracy for different levels of noise (please refer to Table 3.1).

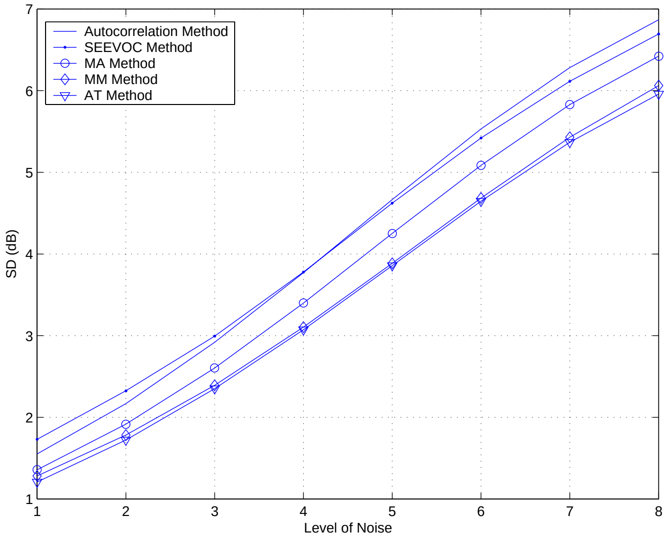

Figure 3.19: Average SD for 12 s of speech spoken by 3 separate subjects introduced with 'restaurant' noise for robust analysis (please refer to Table 3.1).

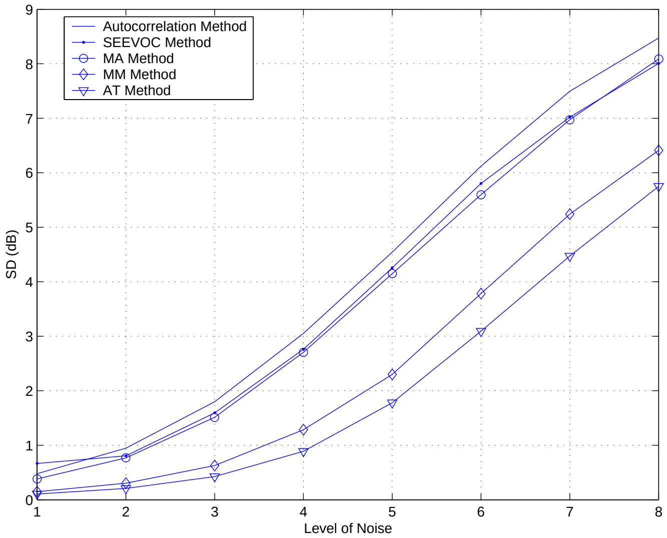

Figure 3.20: Robust SD measurements for 30 ms speech vowel $[e]$ introduced with Gaussian noise (please refer to Table 3.1).

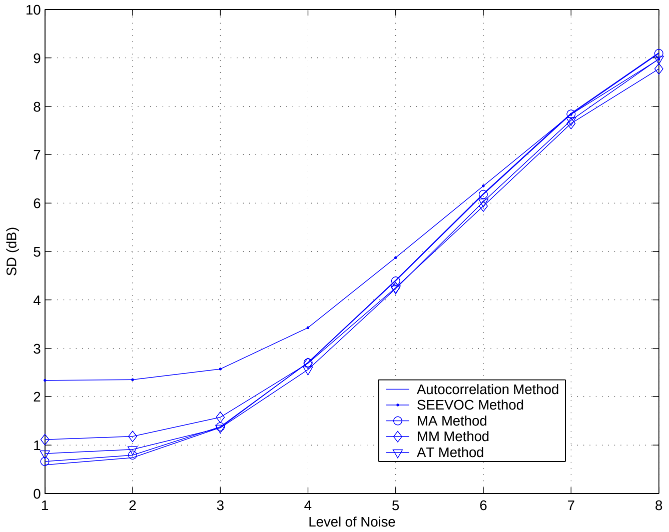

Figure 3.21: Accuracy analysis of speech [e] with added Gaussian noise (please refer to Table 3.1).

Comparison between the robust LP analysis methods can be observed in Figures 3.19, 3.20 and 3.21. The SEEVOC spectrum is computed using CP of 15 frequency samples, with search range [$\frac{1}{2}CP$, $\frac{3}{2}CP$]. The three robust methods apply an analysis window of 49 frequency samples. The AT method is performed with 3 repetitions.

It can be seen that using the above settings, the robust methods perform as accurately as the autocorrelation method of LP analysis. AT method is the most robust method of LP analysis, whilst offering no significant degradation in its accuracy. The simulations performed in this chapter show that the choice of analysis window length affects the robustness and accuracy of the proposed methods, where improvement of one performance aspect will lead to the degradation of the other.

Chapter 4

Quantisation of the LP Parameters

4.1 Scalar Quantisation of LP Parameters

This chapter studies the effect of quantisation on the LP analysis methods. The scalar quantisation (SQ) method using non-uniform levels (as previously explained in Chapter 2) is presented in this section, while quantisation of the parameters using vector quantisation (VQ) will be discussed and simulated in the following section.

It will be shown in this chapter that the proposed methods are better for quantisation in comparison to the conventional methods. It provides less distortion when quantisation is performed on its LP parameters. This allows fewer bits to be allocated during quantisation in order to achieve equal speech quality with the conventional methods.

As explained in Section 2.7.4, non-uniformly-spaced quantisation levels are chosen for simulation of SQ on the LP parameters as it provides a more accurate quantisation, as most signals do not show a uniform behaviour. Non-uniform quantising levels are obtained via the LBG algorithm. The simulations in this chapter were completed using the following settings:

- Codebook train and test procedure uses the *TIMIT* database sampled at 8 kHz. Speech data for the test procedure uses the [*si*] sentences (3 from each speaker) taken from the *TIMIT* test database.
- LP analysis order of 10.
- FFT-computed spectrum length equals 512 frequency samples.
- Frame size set to 20 ms.
- Robust LP analysis window $\simeq$ 330 Hz (21 frequency samples).
- AT repetition of 3 and analysis window size equal to MA.
- For the SEEVOC method, CP set to 10 frequency samples (frequency width $\simeq$ 156 Hz) with search range $[\frac{1}{2}CP,\ \frac{3}{2}CP]$.

The robust LP analysis window lengths and AT repetition constant is determined to provide a good balance between maintaining the accuracy of the LPC spectrum and achieving robustness when noise is introduced to the speech signal. Please refer to Section 3.5 for a more detailed explanation.

The importance of transformation of LP parameters can be seen in Tables 4.1 and 4.2. It can be observed that for the autocorrelation method (AM), an average SD of approximately 1 dB can be reached with 60 quantisation bits, however spectral transparency can only be achieved with 75 bits. A high number of bits for quantisation are also encountered with the other methods. This highlights the need to transform the LP parameters (Section 2.7.4). The LSF transformation is used for the simulations of LP parameter quantisation.

Table 4.1: Performance for quantisation of conventional LP parameters (AM) with no transformation.

Number of	*Average*	*Outliers* (%)	
bits/frame	*SD (dB)*	*2-4 dB*	*>4 dB*
60	1.040	11.351	2.394
65	0.727	6.109	0.920
70	0.576	3.957	0.640
71	0.549	3.634	0.606
72	0.510	3.128	0.475
73	0.468	2.540	0.384
74	0.432	2.067	0.319
75	0.395	1.713	0.260

Table 4.2: Performance for quantisation of LP parameters with no transformation using SEEVOC and proposed methods.

Estimation	*Number of*	*Average*	*Outliers* (%)	
method	*bits/frame*	*SD (dB)*	*2-4 dB*	*>4 dB*
SEEVOC	60	1.022	11.300	2.405
	65	0.703	6.177	0.914
	75	0.373	1.778	0.206
MA	60	1.031	11.326	2.438
	65	0.718	6.231	0.929
	75	0.384	1.720	0.243
MM	60	1.087	12.823	2.716
	65	0.760	7.503	1.159
	75	0.407	2.302	0.291
AT	60	1.026	11.829	2.333
	65	0.702	6.357	0.900
	75	0.374	1.853	0.231

Table 4.3: Quantisation performance of non-uniform SQ using LSF transformation.

Estimation method	*Number of bits/frame*	*Average SD (dB)*	*Outliers (%)*	
			2-4 dB	*>4 dB*
AM	35	1.311	9.192	1.616
	36	1.185	6.691	1.201
	37	1.054	5.153	0.444
	38	0.947	3.774	0.212
	39	0.863	2.576	0.098
	40	0.804	1.807	0.087
SEEVOC	40	0.756	1.595	0.088
MA	40	0.772	1.648	0.090
MM	40	0.759	1.687	0.111
AT	40	0.737	1.542	0.091

The quantisation performance of the autocorrelation (AM), SEEVOC, moving average (MA), moving maximum (MM) and average threshold (AT) methods with LSF transformation can be observed in Table 4.3. The average number of bits needed to achieve spectral transparency is 40 bits/frame, which is a high reduction in comparison to the results in the previous tables.

4.2 Split Vector Quantisation of LP Parameters

Due to computational complexity and added memory space needed for the quantisation codebook, a computationally efficient method is needed for VQ, especially for future speech coding applications. Full-search VQ has a very high computational complexity and requires too much of memory space for the quantisation codebook. The split VQ design is used to investigate the quantisation performance of the LP parameters. Though the split VQ approach is suboptimal, it reduces computational complexity and memory requirements to manageable limits without affecting the

VQ performance too much [84], [85].

As the name suggests, this method divides the LP parameters into separate lower-order partitions. SD calculated from the quantisation of the LP parameters is minimised by searching for the least distortion in smaller-size vector partitions. Separation of the LP parameters is done in its root domain, which is determined to be most accurate, as separation in its time domain would introduce further distortion in its power spectrum.

The separation of the LP polynomials into two partitions (4 LSF's in the first partition and 6 LSF's in the second) is proposed for a 10^{th} order LP analysis [24]. Non-uniform partitioning is favoured to uniform partitioning due to the higher priority of achieving less MSE in the low frequency roots, where the larger bulk of the power spectrum's accuracy is preserved, than in the high frequency roots.

A block diagram of the split VQ procedure for two partitions can be seen in Figure 4.1. In this example, M-dimension quantisation codebook is computed from N input vectors ($N >> M$). Separating the LP parameters to its low and high frequency roots forms each lower-order polynomial. The training of $Q1$ and $Q2$ is done using the LBG algorithm covered in Section 2.4.2. It is clear from this example that training the separate codebook partitions must not be computed from the same set of train vectors, as it must accommodate the splitting of the LP parameters into non-uniform partitions.

By separating the all-pole filter polynomials $A(z)$ into its low and high frequency roots, denoted $L(z)$ and $H(z)$ respectively,

$$\frac{1}{A(z)} = \frac{1}{L(z)}\frac{1}{H(z)} \tag{4.1}$$

the indexes of each codebook can then be determined,

$$j^*, k^* = argminD_{LR}(\frac{1}{A(z)} : \frac{1}{L_j(z)}\frac{1}{H_k(z)}) \tag{4.2}$$

where j^* and k^* are the indices pair for two LP parameter partitions and D_{LR} is the likelihood ratio distortion, which is dictated either by the minimum MSE

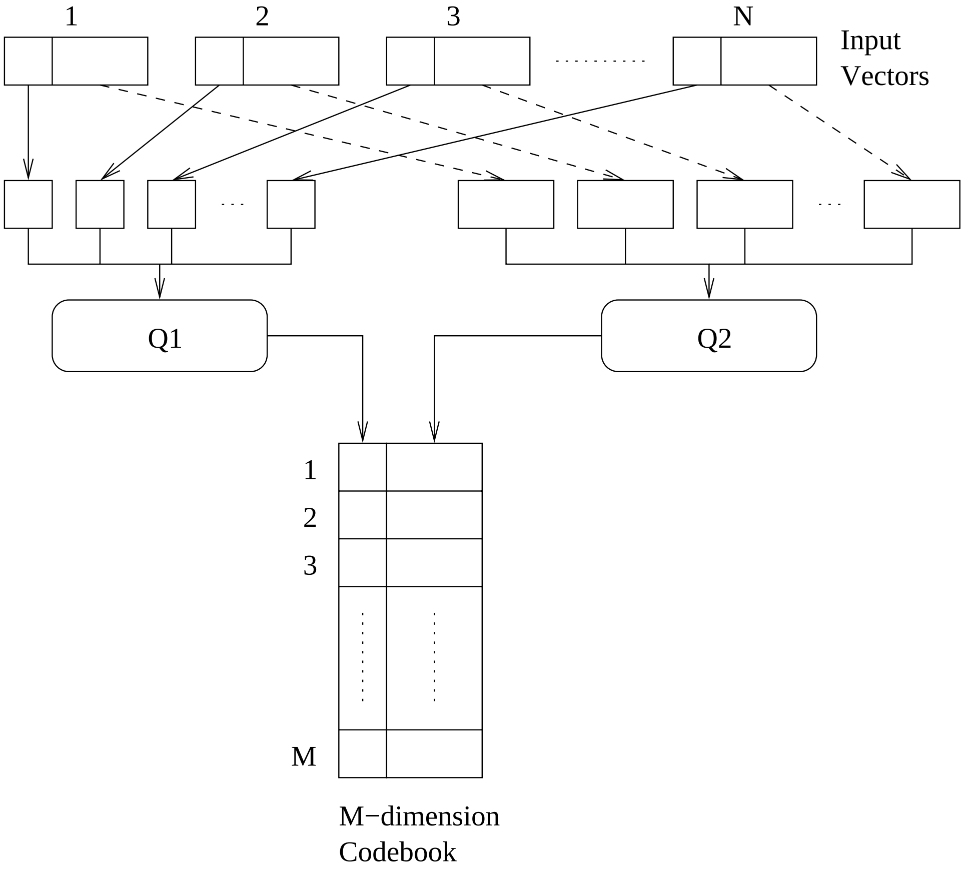

Figure 4.1: Block diagram of the split VQ for 2 partitions.

of each partition or the average SD of the complete set of LP parameters (2.52). By determining the minimum likelihood ratio distortion, the indices pair can be located for each region; $j \; \varepsilon \; 1, 2, \ldots, N_L$ and $k \; \varepsilon \; 1, 2, \ldots, N_H$.

The process of selecting the optimum codebook indices can be achieved by checking every single possible combination from the sets of codebook. Although selecting individual indices by the minimisation of MSE is generally considered accurate for selecting the quantised transformed LP parameters, optimum selection can only be reached by the minimisation of SD. However with the need for large numbers of bit allocated for quantisation (may reach up to 50 bits per 10 coefficients, depending on the quantisation method used), it is computationally expensive to search every index combination.

The computational time for codebook selection can be largely reduced by searching the high frequency codebook using a pre-selected low frequency codebook template and vice versa. Pre-selecting a codebook template is done via individual search of minimum MSE for each codebook index. The difference between selecting the codebook using the minimum MSE and the minimum SD is shown in Table 4.4. As can be seen, only a slight improvement is offered by the minimum SD selection criterion. Further simulations are completed using the minimum MSE selection criterion as it offers much lower computational cost.

Table 4.4: Comparison for quantisation performance with different selection criterion for 3 part split VQ at 18 bits/frame with LSF transformation.

Estimation	*Average*	*Outliers (%)*	
method	*SD (dB)*	*2-4 dB*	*>4 dB*
Minimum MSE selection			
AM	2.005	43.354	0.759
SEEVOC	1.808	29.471	0.314
MA	1.882	35.351	0.404
MM	1.772	27.435	0.368
AT	1.731	24.382	0.259
Minimum SD selection			
AM	1.990	42.478	0.676
SEEVOC	1.800	28.901	0.303
MA	1.874	34.731	0.380
MM	1.766	27.071	0.358
AT	1.725	24.006	0.255

All the simulations in this chapter use LSF transformation with the same settings as in Section 2.4.2. Uniform bit allocation is used for individual parts. Tables 4.5, 4.6 and 4.7 show the quantisation performance in terms of average SD using a split VQ with 2, 3 (3 LSF's in the first partition, 3 LSF's in the second and 4 LSF's in the third) and 5 (2 LSF's in each partition) partitions respectively.

Table 4.5: Quantisation performance of different methods for 2 part split VQ at 24 bits/frame.

Estimation	*Average*	*Outliers (%)*	
method	*SD (dB)*	*2-4 dB*	*>4 dB*
AM	1.374	9.088	0.023
SEEVOC	1.220	3.823	0.009
MA	1.276	5.318	0.013
MM	1.186	3.236	0.009
AT	1.154	2.344	0.002

Table 4.6: Quantisation performance of different methods for 3 part split VQ.

Estimation	*Number of*	*Average*	*Outliers (%)*	
method	*bits/frame*	*SD (dB)*	*2-4 dB*	*>4 dB*
AM	26	1.281	6.063	0.027
	27	1.164	3.223	0.014
SEEVOC	26	1.150	2.881	0.022
	27	1.042	1.291	0.008
MA	26	1.197	3.680	0.021
	27	1.084	1.826	0.001
MM	26	1.126	2.485	0.021
	27	1.014	1.116	0.006
AT	26	1.095	1.875	0.016
	27	0.986	0.751	0.003

Figure 4.2 shows the average SD for VQ with no partition (10^{th} order LP analysis) for varying bits/frame. It can be seen from these simulation results that all the robust methods provide better quantisation performance than the autocorrelation

Table 4.7: Quantisation performance of different methods for 5 part split VQ at 30 bits/frame.

Estimation method	*Average SD (dB)*	*Outliers (%)*	
		2-4 dB	*>4 dB*
AM	1.073	1.781	0.022
SEEVOC	0.976	0.926	0.020
MA	1.006	1.082	0.018
MM	0.967	1.066	0.023
AT	0.930	0.716	0.014

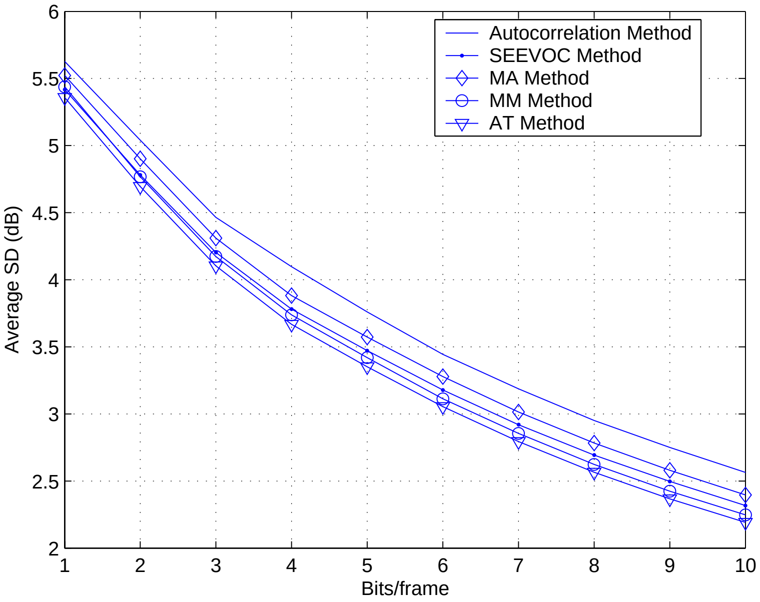

Figure 4.2: Average SD for VQ with no partition.

method. The AT method offers the best performance from all the robust methods.

An improvement of 2-3 bits/frame is observed when comparing the AT method to the conventional autocorrelation method (AM) of LP analysis. Referring to

Table 4.8: Performance of the conventional LP analysis methods for 3 part split VQ.

Estimation method	*Number of bits/frame*	*Average SD (dB)*	*Outliers (%)*	
			2-4 dB	*>4 dB*
AM	23	1.531	15.124	0.094
	24	1.396	8.929	0.032
	25	1.357	8.126	0.029
	26	1.281	6.063	0.027
	27	1.164	3.223	0.014
SEEVOC	25	1.220	3.900	0.026
	26	1.150	2.881	0.022
	27	1.042	1.291	0.008

Table 4.9: Performance of the robust LP analysis methods for 3 part split VQ.

Estimation method	*Number of bits/frame*	*Average SD (dB)*	*Outliers (%)*	
			2-4 dB	*>4 dB*
MA	25	1.270	5.000	0.030
	26	1.197	3.680	0.021
	27	1.084	1.826	0.001
MM	25	1.197	3.423	0.022
	26	1.126	2.485	0.021
	27	1.014	1.116	0.006
AT	21	1.433	8.983	0.083
	22	1.399	8.263	0.083
	23	1.315	5.771	0.077
	24	1.190	2.765	0.019
	25	1.164	2.560	0.018
	26	1.095	1.875	0.016
	27	0.986	0.751	0.003

Tables 4.8 and 4.9, for 3 part split VQ, an example of 23 bits/frame for the AT method would result in an average SD that would resemble that of the AM method with 25 bits/frame. It should be noted also that for the AT method with 23 bits/frame, a significantly less percentage of outlier frames is observed.

Chapter 5

Low Bit-Rate Speech Coding Application

5.1 Application of the Robust LP Analysis Methods in CELP

The performance of the robust methods for quantisation in low bit-rate speech coding application is investigated in this chapter. The CELP coder, previously defined in Section 2.7, is used to simulate the performance of the robust methods.

The settings for the simulations performed in this chapter follow that of the previous chapter (Section 4.1). The performance criterion for the CELP coder is performed using the Segmental-SNR, defined in Section 2.8.3, and will be further represented by the term SNR. Quantisation of the LP parameters is performed on the LSF coefficients using the split VQ approach. The simulation settings for the CELP coder are as follows:

- LP analysis order of 10 on a 20 ms frame.

- Pitch delay of 128 samples.
- 7-bit random Gaussian codebook used for determining the excitation parameters.

Simulations in this chapter are performed on two separate sets of sentences taken from the *TIMIT* test database, described as follows:

- **Set 0**: compilation of 40 sentences spoken by 38 separate speakers (23 male speakers and 15 female speakers) randomly selected from the *TIMIT* test database. This set of sentences contains 123.24 seconds of speech (6162 frames of 20 ms length of speech) consisting of 61.6 seconds (3080 frames) of non-silent speech.
- **Set 1**: 40 sentences spoken by 2 male and 2 female speakers, each reading 10 sentences, selected from the test database. This set contains of 6815 frames of speech (136.3 seconds) with 3578 frames of non-silent speech.

The simulations performed in this chapter compare the moving maximum (MM), moving average (MA) and average threshold (AT) methods to the conventional autocorrelation (AM) and SEEVOC methods. The SEEVOC method is simulated using a CP of 10 frequency samples (frequency width $\simeq$ 156 Hz) with search range $[\frac{1}{2}CP, \frac{3}{2}CP]$.

Tables 5.1 to 5.3 show the performance of the robust LP analysis methods. For 5 part (30 bits/frame) and 2 part (24 bits/frame) split VQ, the robust methods are shown to perform better than the AM method in terms of speech quality. Although the MA method is shown to be most affected by quantisation in comparison to the MM and AT method (in terms of SD), it offers the highest SNR for both sets of sentences.

Simulations using various number of bits/frame for the 3 part split VQ also show identical behaviour to the above case. The conventional LP analysis methods are

Table 5.1: CELP performance for the different LP analysis methods in SNR (dB).

Set 0			
Estimation method	*Without quantisation*	*5 part (30 bits/frame)*	*2 part (24 bits/frame)*
AM	10.37	9.59	9.68
SEEVOC	9.97	9.55	9.55
MM	9.91	9.52	9.56
MA	10.57	9.86	9.93
AT	10.31	9.87	9.94
Set 1			
Estimation method	*Without quantisation*	*5 part (30 bits/frame)*	*2 part (24 bits/frame)*
AM	10.14	9.42	9.53
SEEVOC	9.95	9.47	9.53
MM	9.65	9.30	9.35
MA	10.38	9.69	9.77
AT	10.10	9.65	9.72

Table 5.2: CELP performance for 3 part split VQ on Set 0 sentences in SNR (dB).

Estimation method	*Number of bits/frame*			
	18	*21*	*24*	*27*
AM	9.21	9.42	9.65	9.90
SEEVOC	9.17	9.39	9.53	9.68
MM	9.19	9.40	9.59	9.67
MA	9.40	9.73	9.91	10.04
AT	9.48	9.71	9.89	10.01

most affected by quantisation, as the degradation of its SNR can be seen as the number of bits/frame is decreased from 27 to 18 bits/frame. Also the MM and AT

Table 5.3: CELP performance for 3 part split VQ on Set 1 sentences in SNR (dB).

Estimation	*Number of bits/frame*			
method	*18*	*21*	*24*	*27*
AM	9.04	9.36	9.48	9.68
SEEVOC	9.07	9.32	9.51	9.63
MM	9.09	9.29	9.40	9.51
MA	9.31	9.55	9.74	9.90
AT	9.39	9.54	9.73	9.85

methods offer the most robust methods as the quantisation bits are varied for the 3 part split VQ.

5.2 Noise Introduction

5.2.1 Real World Noise

The introduction of real world noise into the speech samples is simulated using the *babble*, *street* and *restaurant* noise samples from the *Aurora* database. Tables 5.4 and 5.5 show the performance of all the methods for speech affected by babble noise with quantisation using 3 part split VQ.

For speech affected by low-level noise, the MM and SEEVOC methods show a slightly lower SNR value to the AM method. The level of noise is calculated using SNR and determined from the clean speech against the speech affected by noise. However as the effect of noise is increased, the SEEVOC and proposed methods show a more robust behaviour than the AM method. The proposed methods are also observed to produce a higher SNR for high-levels of noise. Also the proposed methods outperform both the AM and SEEVOC methods in terms of robustness

Table 5.4: Performance for 3 part split VQ on Set 0 sentences with babble noise in SNR (dB).

Estimation	*Noise level*	*Bits/frame*		
method	*(SNR)*	*18*	*21*	*24*
AM	35 dB	9.17	9.43	9.61
	30 dB	9.14	9.43	9.61
	25 dB	9.01	9.27	9.46
	5 dB	6.28	6.34	6.44
	0 dB	5.61	5.72	5.79
SEEVOC	35 dB	9.14	9.33	9.55
	30 dB	9.13	9.31	9.43
	25 dB	8.97	9.16	9.25
	5 dB	6.32	6.40	6.46
	0 dB	5.64	5.74	5.76
MM	35 dB	9.15	9.40	9.58
	30 dB	9.14	9.39	9.55
	25 dB	9.08	9.30	9.43
	5 dB	6.49	6.58	6.64
	0 dB	5.79	5.90	5.93
MA	35 dB	9.37	9.65	9.86
	30 dB	9.36	9.61	9.85
	25 dB	9.22	9.53	9.73
	5 dB	6.44	6.55	6.65
	0 dB	5.74	5.83	5.90
AT	35 dB	9.46	9.74	9.87
	30 dB	9.46	9.64	9.85
	25 dB	9.38	9.54	9.74
	5 dB	6.54	6.63	6.73
	0 dB	5.83	5.89	5.96

Table 5.5: Performance for 3 part split VQ on Set 1 sentences with babble noise in SNR (dB).

Estimation method	*Noise level (SNR)*	*Bits/frame*		
		18	*21*	*24*
AM	35 dB	9.03	9.30	9.46
	30 dB	9.00	9.30	9.44
	25 dB	8.90	9.16	9.27
	5 dB	6.21	6.36	6.42
	0 dB	5.57	5.68	5.75
SEEVOC	35 dB	9.10	9.29	9.44
	30 dB	9.07	9.27	9.38
	25 dB	8.92	9.12	9.22
	5 dB	6.25	6.36	6.43
	0 dB	5.60	5.69	5.73
MM	35 dB	9.06	9.30	9.38
	30 dB	9.05	9.23	9.35
	25 dB	8.95	9.12	9.22
	5 dB	6.46	6.50	6.56
	0 dB	5.75	5.83	5.87
MA	35 dB	9.28	9.52	9.73
	30 dB	9.27	9.53	9.67
	25 dB	9.12	9.36	9.51
	5 dB	6.35	6.49	6.59
	0 dB	5.69	5.79	5.83
AT	35 dB	9.37	9.55	9.73
	30 dB	9.32	9.50	9.69
	25 dB	9.19	9.40	9.56
	5 dB	6.44	6.57	6.63
	0 dB	5.77	5.83	5.89

for applications in speech affected by high-levels of noise.

As the SNR measurement tool is not the most accurate quality measure for speech coding, the quality of the synthesised speech signal needs to be determined by a human subject listening directly to the speech. Through this process, it is evident that for 18 bits/frame there exists a difference in speech quality between the AM method and the robust methods. Although this cannot be clearly observed throughout the two speech sets, there are certain sections of the speech where the quality of speech from the AM method is not as clear as the speech from the robust methods. This observation was obvious for high levels of noise.

In the cases where the differences in quality are obvious (for example, when speech is affected by high-level babble noise (5 dB SNR)), the synthesised speech from the AM method cannot be clearly observed over the noise. Using the robust methods, the speech outputs are observed to be less distorted, with the actual speech observed slightly more clearly over noise.

Tables 5.6 to 5.8 show the performance of the LP analysis methods for various split VQ partition settings when speech is affected by various real world background noise samples. It can be observed that although the SEEVOC method provides a slight improvement to the conventional AM method, the robust methods offer significantly better improvement to the robustness (when speech is introduced into noisy environments) and synthesised speech quality. In general, the AT method provides the best SNR improvement, approximately 0.4 dB for varying speech samples, followed by the MM and MA methods.

Table 5.6: Performance for 5 (30 bits/frame) and 2 (24 bits/frame) part split VQ with babble noise in SNR (dB).

Estimation method	*Noise level (SNR)*	*5 part*		*2 part*	
		Set 0	*Set 1*	*Set 0*	*Set 1*
AM	35 dB	9.59	9.41	9.68	9.50
	30 dB	9.57	9.39	9.67	9.46
	25 dB	9.45	9.26	9.53	9.29
	5 dB	6.51	6.44	6.53	6.45
	0 dB	5.82	5.77	5.81	5.79
SEEVOC	35 dB	9.54	9.41	9.54	9.48
	30 dB	9.42	9.36	9.50	9.41
	25 dB	9.28	9.21	9.34	9.26
	5 dB	6.45	6.42	6.49	6.44
	0 dB	5.80	5.74	5.81	5.74
MM	35 dB	9.51	9.31	9.56	9.33
	30 dB	9.48	9.25	9.53	9.30
	25 dB	9.41	9.15	9.44	9.21
	5 dB	6.68	6.54	6.63	6.60
	0 dB	5.91	5.87	5.94	5.89
MA	35 dB	9.87	9.70	9.91	9.75
	30 dB	9.83	9.63	9.89	9.67
	25 dB	9.71	9.48	9.79	9.56
	5 dB	6.66	6.58	6.67	6.58
	0 dB	5.92	5.88	5.93	5.88
AT	35 dB	9.86	9.62	9.89	9.72
	30 dB	9.78	9.58	9.85	9.66
	25 dB	9.73	9.46	9.76	9.53
	5 dB	6.75	6.63	6.77	6.64
	0 dB	5.98	5.90	5.98	5.91

Table 5.7: Performance for 3 part split VQ at 27 bits/frame with various real world noise samples introduced to Set 0 sentences in SNR (dB).

Babble						
Estimation method	*Noise level (SNR)*					
	35 dB	*30 dB*	*25 dB*	*20 dB*	*15 dB*	*5 dB*
AM	9.80	9.79	9.63	9.28	8.58	6.56
SEEVOC	9.62	9.57	9.42	9.08	8.41	6.52
MM	9.65	9.64	9.53	9.27	8.62	6.69
MA	10.03	10.03	9.87	9.54	8.81	6.67
AT	10.01	9.93	9.84	9.48	8.84	6.78

Street						
Estimation method	*Noise level (SNR)*					
	35 dB	*30 dB*	*25 dB*	*20 dB*	*15 dB*	*5 dB*
AM	9.87	9.80	9.53	8.99	7.89	4.56
SEEVOC	9.67	9.60	9.40	8.85	7.80	4.59
MM	9.69	9.67	9.49	9.01	8.04	4.72
MA	10.08	10.04	9.79	9.22	8.10	4.65
AT	10.04	9.99	9.75	9.26	8.13	4.74

Restaurant						
Estimation method	*Noise level (SNR)*					
	35 dB	*30 dB*	*25 dB*	*20 dB*	*15 dB*	*5 dB*
AM	9.82	9.80	9.67	9.30	8.56	6.17
SEEVOC	9.64	9.63	9.47	9.13	8.45	6.18
MM	9.65	9.63	9.54	9.23	8.66	6.36
MA	10.00	10.02	9.88	9.50	8.77	6.32
AT	10.01	9.98	9.86	9.53	8.78	6.39

Table 5.8: Performance for 3 part split VQ at 27 bits/frame with various real world noise samples introduced to Set 1 sentences in SNR (dB).

Babble						
Estimation method	*Noise level (SNR)*					
	35 dB	*30 dB*	*25 dB*	*20 dB*	*15 dB*	*5 dB*
AM	9.66	9.59	9.39	9.05	8.41	6.47
SEEVOC	9.58	9.53	9.35	8.96	8.33	6.45
MM	9.47	9.43	9.32	9.04	8.42	6.60
MA	9.87	9.81	9.65	9.29	8.59	6.61
AT	9.82	9.77	9.67	9.31	8.65	6.66
Street						
Estimation method	*Noise level (SNR)*					
	35 dB	*30 dB*	*25 dB*	*20 dB*	*15 dB*	*5 dB*
AM	9.64	9.55	9.34	8.79	7.76	4.58
SEEVOC	9.61	9.51	9.30	8.78	7.75	4.64
MM	9.48	9.48	9.25	8.75	7.81	4.73
MA	9.86	9.81	9.59	9.00	7.94	4.69
AT	9.79	9.76	9.55	9.00	7.96	4.75
Restaurant						
Estimation method	*Noise level (SNR)*					
	35 dB	*30 dB*	*25 dB*	*20 dB*	*15 dB*	*5 dB*
AM	9.66	9.60	9.44	9.11	8.37	6.07
SEEVOC	9.58	9.51	9.36	8.99	8.33	6.11
MM	9.46	9.41	9.32	9.03	8.38	6.25
MA	9.91	9.82	9.68	9.26	8.57	6.21
AT	9.83	9.79	9.62	9.28	8.60	6.29

5.2.2 Gaussian Noise

Table 5.9: Performance for 3 part split VQ at 18 bits/frame with Gaussian noise on Set 0 in SNR (dB).

Estimation	*Noise level (SNR)*					
method	*35 dB*	*30 dB*	*25 dB*	*20 dB*	*15 dB*	*5 dB*
AM	9.21	9.18	9.01	8.49	7.44	4.38
SEEVOC	9.20	9.12	8.99	8.50	7.54	4.44
MM	9.24	9.17	9.05	8.58	7.62	4.57
MA	9.43	9.36	9.21	8.71	7.65	4.46
AT	9.53	9.46	9.29	8.80	7.80	4.56

Table 5.10: Performance for 3 part split VQ at 18 bits/frame with Gaussian noise on Set 1 in SNR (dB).

Estimation	*Noise level (SNR)*					
method	*35 dB*	*30 dB*	*25 dB*	*20 dB*	*15 dB*	*5 dB*
AM	9.03	8.99	8.78	8.32	7.41	4.39
SEEVOC	9.08	9.02	8.80	8.34	7.43	4.52
MM	9.13	9.09	8.91	8.45	7.55	4.60
MA	9.31	9.21	8.99	8.52	7.56	4.50
AT	9.36	9.28	9.12	8.60	7.65	4.61

Performance of the robust methods when introduced with Gaussian noise can be observed in Tables 5.9 to 5.14. As observed in the previous section, the AT method provides the most robust performance. The improvement in speech quality of the robust methods compared to the conventional methods is more apparent for speech affected by high-levels of noise. This is more obvious to the human ear for SNR of approximately 10 to -5 dB (where the noise is generally at the same amplitude level as the clean speech).

When listening to the speech outputs, it is apparent that the SEEVOC method does

Table 5.11: Performance for 3 part split VQ with Gaussian noise on Set 0 in SNR (dB).

Estimation method	*Number of bits/frame*	*Noise level (SNR)*			
		35 dB	*25 dB*	*15 dB*	*5 dB*
AM	21	9.40	9.22	7.62	4.46
	24	9.66	9.39	7.78	4.54
	27	9.87	9.53	7.93	4.57
SEEVOC	21	9.38	9.19	7.63	4.48
	24	9.57	9.31	7.73	4.54
	27	9.66	9.43	7.76	4.58
MM	21	9.47	9.26	7.86	4.65
	24	9.59	9.38	7.95	4.70
	27	9.67	9.50	8.00	4.72
MA	21	9.72	9.50	7.82	4.55
	24	9.90	9.66	7.98	4.62
	27	10.08	9.83	8.09	4.67
AT	21	9.75	9.46	7.95	4.65
	24	9.92	9.66	8.08	4.70
	27	10.02	9.81	8.15	4.73

not always provide an improvement to the AM method. For high levels of noise the synthesised speech calculated from the SEEVOC method cannot be clearly observed over the noise, especially in comparison to the speech calculated from the AM method. It is noted that the AT and MM methods provide speech outputs with less audible treble, which provides better speech quality.

Table 5.12: Performance for 3 part split VQ with Gaussian noise on Set 1 in SNR (dB).

Estimation	*Number of*	*Noise level (SNR)*			
method	*bits/frame*	*35 dB*	*25 dB*	*15 dB*	*5 dB*
AM	21	9.33	9.06	7.54	4.46
	24	9.50	9.16	7.63	4.56
	27	9.66	9.37	7.76	4.58
SEEVOC	21	9.32	9.01	7.55	4.54
	24	9.46	9.17	7.70	4.61
	27	9.55	9.32	7.75	4.64
MM	21	9.28	9.03	7.67	4.67
	24	9.38	9.17	7.77	4.74
	27	9.46	9.23	7.81	4.76
MA	21	9.50	9.24	7.72	4.58
	24	9.72	9.42	7.83	4.65
	27	9.88	9.56	7.96	4.69
AT	21	9.58	9.31	7.79	4.65
	24	9.73	9.42	7.93	4.70
	27	9.86	9.56	8.01	4.76

Table 5.13: Performance for 2 part split VQ (24 bits/frame) with Gaussian noise on Set 0 in SNR (dB).

Estimation	*Set 0*			*Set 1*		
method	*35 dB*	*25 dB*	*15 dB*	*35 dB*	*25 dB*	*15 dB*
AM	9.73	9.48	7.86	9.55	9.24	7.71
SEEVOC	9.54	9.35	7.79	9.51	9.23	7.72
MM	9.58	9.43	8.01	9.35	9.17	7.81
MA	9.98	9.73	8.06	9.78	9.50	7.90
AT	9.95	9.75	8.13	9.70	9.46	7.95

Table 5.14: Performance for 5 part split VQ (30 bits/frame) with Gaussian noise on Set 0 in SNR (dB).

Estimation	*Set 0*			*Set 1*		
method	*35 dB*	*25 dB*	*15 dB*	*35 dB*	*25 dB*	*15 dB*
AM	9.66	9.51	7.91	9.44	9.19	7.71
SEEVOC	9.50	9.31	7.81	9.46	9.17	7.70
MM	9.54	9.38	7.98	9.32	9.09	7.77
MA	9.91	9.65	8.05	9.68	9.42	7.89
AT	9.92	9.68	8.08	9.64	9.44	7.92

5.3 Variation of the Analysis Window Lengths

Table 5.15: SNR performance comparison for 3 part split VQ (18 bits/frame) with varying analysis window lengths and no noise on Set 0 sentences in SNR (dB).

Estimation	*Average*	*Outliers (%)*		*SNR*
method	*SD (dB)*	*2-4 dB*	*>4 dB*	*(dB)*
M = 21 frequency samples				
MM	1.812	30.867	0.552	9.19
MA	1.917	36.952	0.406	9.40
AT	1.765	26.874	0.389	9.48
M = 7 frequency samples				
MM	1.993	43.411	0.552	9.28
MA	2.031	44.677	0.957	9.15
AT	1.993	43.395	0.471	9.29

The effect of varying the analysis window size of the robust methods is investigated here. The performance of the robust methods is compared for analysis window lengths of 21 and 7 frequency samples on the FFT-computed spectrum, translating to approximately 330 and 110 Hz respectively for an 8 kHz speech signal. Comparison of performance for clean speech can be observed for 3 part (18

Table 5.16: SNR performance comparison for 3 part split VQ (18 bits/frame) with varying analysis window lengths and no noise on Set 1 sentences in SNR (dB).

Estimation method	*Average SD (dB)*	*Outliers (%)*		*SNR (dB)*
		2-4 dB	*>4 dB*	
M = 21 frequency samples				
MM	1.732	24.519	0.147	9.09
MA	1.816	30.858	0.117	9.31
AT	1.673	20.807	0.088	9.39
M = 7 frequency samples				
MM	1.892	36.067	0.279	9.16
MA	1.938	38.665	0.572	9.06
AT	1.905	37.080	0.264	9.11

Table 5.17: SNR performance comparison for 3 part split VQ (21 bits/frame) with varying analysis window lengths and no noise on Set 0 sentences in SNR (dB).

Estimation method	*Average SD (dB)*	*Outliers (%)*		*SNR (dB)*
		2-4 dB	*>4 dB*	
M = 21 frequency samples				
MM	1.504	12.691	0.243	9.40
MA	1.597	17.218	0.195	9.73
AT	1.464	10.565	0.179	9.71
M = 7 frequency samples				
MM	1.656	20.480	0.211	9.54
MA	1.698	23.239	0.097	9.43
AT	1.664	22.233	0.097	9.58

and 21 bits/frame) and 5 part (30 bits/frame) split VQ in Tables 5.15 to 5.20. Tables 5.21 to 5.24 show the performance when babble and street noise is introduced to 3 part split VQ.

Table 5.18: SNR performance comparison for 3 part split VQ (21 bits/frame) with varying analysis window lengths and no noise on Set 1 sentences in SNR (dB).

Estimation method	*Average SD (dB)*	*Outliers (%)*		*SNR (dB)*
		2-4 dB	*>4 dB*	
M = 21 frequency samples				
MM	1.435	9.171	0.059	9.29
MA	1.512	11.930	0.015	9.55
AT	1.387	6.750	0.015	9.54
M = 7 frequency samples				
MM	1.575	15.671	0.000	9.45
MA	1.938	38.665	0.572	9.06
AT	1.905	37.080	0.264	9.11

Table 5.19: SNR performance comparison for 5 part split VQ (30 bits/frame) with varying analysis window lengths and no noise on Set 0 sentences in SNR (dB).

Estimation method	*Average SD (dB)*	*Outliers (%)*		*SNR (dB)*
		2-4 dB	*>4 dB*	
M = 21 frequency samples				
MM	0.982	1.331	0.000	9.52
MA	1.029	1.525	0.000	9.86
AT	0.950	0.828	0.016	9.87
M = 7 frequency samples				
MM	1.065	1.866	0.049	9.72
MA	1.090	2.272	0.016	9.65
AT	1.069	1.736	0.049	9.77

Narrow analysis windows are more effective in allocating the spectral peaks of the FFT-computed power spectrum, thus providing a more accurate LPC spectrum. However, as investigated in Section 3.5, its robustness suffers when introduced into noisy environments. Comparing the MM method to the conventional LP

Table 5.20: SNR performance comparison for 5 part split VQ (30 bits/frame) with varying analysis window lengths and no noise on Set 1 sentences in SNR (dB).

Estimation	*Average*	*Outliers (%)*		*SNR*
method	*SD (dB)*	*2-4 dB*	*>4 dB*	*(dB)*
M = 21 frequency samples				
MM	0.960	0.939	0.029	9.30
MA	1.002	1.012	0.000	9.69
AT	0.920	0.528	0.000	9.65
M = 7 frequency samples				
MM	1.041	1.467	0.000	9.59
MA	1.062	1.790	0.000	9.46
AT	1.045	1.394	0.000	9.55

analysis methods (Tables 5.1 to 5.3), an improvement of approximately 0.2 dB can be observed for 5 and 3 part split VQ. Although the MM method with narrow analysis windows is not as robust for speech affected by high-levels of noise, the quality of speech is still comparable to the AM method.

The robustness of the MA and AT methods is not significantly affected for narrow analysis windows, although a slight improvement can be seen. However it is noted that quantisation of clean speech without noise for narrow analysis windows results in a slightly lower quality speech.

Table 5.21: SNR performance comparison for 3 part split VQ (18 bits/frame) with varying analysis window lengths and babble noise in SNR (dB).

Estimation method	*Noise level*	*M = 21*		*M = 7*	
		SD (dB)	*SNR (dB)*	*SD (dB)*	*SNR (dB)*
Set 0					
MM	30 dB	2.812	9.14	2.988	9.28
	20 dB	3.659	8.79	3.826	8.86
	10 dB	4.939	7.38	5.090	7.31
	0 dB	6.309	5.79	6.454	5.69
	-10 dB	7.304	5.53	7.447	5.41
MA	30 dB	2.905	9.36	3.033	9.12
	20 dB	3.754	8.94	3.876	8.74
	10 dB	5.023	7.42	5.146	7.23
	0 dB	6.384	5.74	6.519	5.60
	-10 dB	7.370	5.44	7.497	5.37
AT	30 dB	2.743	9.46	2.972	9.21
	20 dB	3.583	9.04	3.814	8.82
	10 dB	4.844	7.53	5.073	7.33
	0 dB	6.225	5.83	6.445	5.68
	-10 dB	7.198	5.54	7.436	5.38
Set 1					
MM	35 dB	2.614	9.06	2.779	9.17
	30 dB	2.957	9.05	3.125	9.12
	-10 dB	7.443	5.51	7.604	5.38
MA	35 dB	2.686	9.28	2.832	9.12
	30 dB	3.029	9.27	3.172	9.02
	-10 dB	7.516	5.41	7.650	5.33
AT	35 dB	2.522	9.37	2.793	9.09
	30 dB	2.868	9.32	3.128	9.07
	-10 dB	7.340	5.48	7.599	5.35

Table 5.22: SNR performance comparison for 3 part split VQ (18 bits/frame) with varying analysis window lengths and street noise in SNR (dB).

Estimation method	*Noise level*	*M = 21*		*M = 7*	
		SD (dB)	*SNR (dB)*	*SD (dB)*	*SNR (dB)*
Set 0					
MM	35 dB	2.901	9.19	3.056	9.31
	30 dB	3.390	9.18	3.548	9.23
	0 dB	8.254	3.14	8.258	3.10
	-5 dB	8.822	2.28	8.746	2.23
	-10 dB	9.166	1.87	9.055	1.85
MA	35 dB	3.028	9.44	3.110	9.15
	30 dB	3.540	9.34	3.623	9.15
	0 dB	8.285	3.08	8.287	3.05
	-5 dB	8.761	2.23	8.758	2.21
	-10 dB	9.054	1.85	9.044	1.83
AT	35 dB	2.888	9.53	3.055	9.23
	30 dB	3.398	9.44	3.547	9.15
	0 dB	8.236	3.14	8.230	3.07
	-5 dB	8.745	2.25	8.695	2.22
	-10 dB	9.057	1.86	8.979	1.84
Set 1					
MM	35 dB	2.795	9.11	2.898	9.21
	30 dB	3.198	9.07	3.309	9.17
	-10 dB	8.336	1.88	8.177	1.85
MA	35 dB	2.869	9.31	2.948	9.10
	30 dB	3.300	9.21	3.370	9.05
	-10 dB	8.179	1.85	8.138	1.83
AT	35 dB	2.749	9.35	2.895	9.14
	30 dB	3.175	9.33	3.311	9.05
	-10 dB	8.209	1.86	8.105	1.85

Table 5.23: SNR performance comparison for 3 part split VQ (21 bits/frame) with varying analysis window lengths and babble noise in SNR (dB).

Estimation method	*Noise level*	*M = 21*		*M = 7*	
		SD (dB)	*SNR (dB)*	*SD (dB)*	*SNR (dB)*
Set 0					
MM	35 dB	2.259	9.40	2.411	9.54
	30 dB	2.575	9.39	2.731	9.50
	-5 dB	6.846	5.59	4.973	5.52
	-10 dB	7.268	5.58	7.408	5.53
MA	35 dB	2.335	9.65	2.454	9.40
	30 dB	2.648	9.61	2.768	9.43
	-5 dB	6.908	5.53	7.033	5.48
	-10 dB	7.345	5.52	7.460	5.47
AT	35 dB	2.195	9.74	2.414	9.53
	30 dB	2.512	9.64	2.727	9.53
	-5 dB	6.747	5.60	6.962	5.49
	-10 dB	7.182	5.58	7.396	5.49
Set 1					
MM	35 dB	2.375	9.30	2.524	9.45
	30 dB	2.742	9.23	2.885	9.41
	-5 dB	6.948	5.52	7.108	5.47
	-10 dB	7.415	5.54	7.569	5.50
MA	35 dB	2.437	9.52	2.574	9.31
	30 dB	2.805	9.53	2.933	9.28
	-5 dB	7.040	5.52	7.171	5.41
	-10 dB	7.498	5.51	7.621	5.44
AT	35 dB	2.293	9.55	2.522	9.43
	30 dB	2.660	9.50	2.883	9.40
	-5 dB	6.861	5.53	7.112	5.46
	-10 dB	7.321	5.54	7.559	5.48

Table 5.24: SNR performance comparison for 3 part split VQ (21 bits/frame) with varying analysis window lengths and street noise in SNR (dB).

Estimation method	*Noise level*	*M = 21*		*M = 7*	
		SD (dB)	*SNR (dB)*	*SD (dB)*	*SNR (dB)*
Set 0					
MM	35 dB	2.683	9.45	2.814	9.55
	30 dB	3.217	9.46	3.355	9.54
	-5 dB	8.788	2.31	8.726	2.25
	-10 dB	9.141	1.90	9.026	1.87
MA	35 dB	2.798	9.73	2.885	9.41
	30 dB	3.359	9.65	3.430	9.37
	-5 dB	8.737	2.25	8.718	2.24
	-10 dB	9.034	1.87	9.010	1.86
AT	35 dB	2.682	9.75	2.818	9.55
	30 dB	3.230	9.73	3.364	9.54
	-5 dB	8.714	2.28	8.660	2.25
	-10 dB	9.044	1.88	8.953	1.86
Set 1					
MM	35 dB	2.588	9.27	2.673	9.42
	30 dB	3.019	9.23	3.107	9.35
	-5 dB	7.920	2.33	7.830	2.27
	-10 dB	8.306	1.90	8.147	1.87
MA	35 dB	2.651	9.54	2.713	9.34
	30 dB	3.107	9.47	3.168	9.20
	-5 dB	7.844	2.27	7.794	2.25
	-10 dB	8.152	1.87	8.099	1.85
AT	35 dB	2.551	9.55	2.663	9.45
	30 dB	2.999	9.50	3.107	9.30
	-5 dB	7.846	2.30	7.761	2.26
	-10 dB	8.185	1.88	8.061	1.86

Chapter 6

Conclusions

6.1 Summary

This dissertation deals with robust LP analysis of speech in noisy environments. Three robust LP analysis methods have been investigated for applications in low bit-rate speech coding. A common approach in the design of the proposed methods is undertaken. It manipulates the FFT-computed power spectrum of a signal in order to remove the effect of noise in speech. This is achieved by allocating more attention on the spectral peaks of the power spectrum, which are least affected by the introduction of noise.

The three robust LP analysis methods, which are the moving maximum (MM), moving average (MA) and average threshold (AT) methods, were investigated and compared to the conventional autocorrelation and SEEVOC methods. These methods improve the robustness of the conventional methods when speech signals are introduced into noisy environments. Also, they provide less quantisation distortion for applications in speech coding.

6.1.1 Observations on Robustness and Accuracy

The robust LP analysis methods were introduced and explained in Chapter 3. The performances of its robustness and accuracy were further investigated in Section 3.5. With an FFT-computed power spectrum length of 512 frequency samples, the simulations were performed for varying analysis window lengths, from approximately 50 to 1600 Hz (ranging from 3 to 101 frequency samples for an 8 kHz speech signal), and the robust spectral analysis methods were applied on speech with varying levels of real world noise.

It was shown for the MA method that the robustness is compensated by its accuracy. As the analysis window length was increased, a steady improvement in its robustness was observed. Narrow analysis windows (between 3 and 10 frequency samples) do not show much variation in its accuracy, however an increase of window length larger than 10 frequency samples would result in a linear decrease in its accuracy.

The MM method shows a similar behaviour to the MA method. Its robustness slightly improves as the MM analysis window length is increased. As with the MA method, this method shows a much better rate of robustness as more noise is introduced to the speech (Figures 3.10 and 3.12). Simulation on its accuracy however shows a similar behaviour to the MA method. However, the decrease rate of its accuracy (as the window length is increased) is much higher for shorter window lengths (below 50 frequency samples) than for larger window lengths (as shown in Figure 3.13).

The simulation for the AT method shows the effect of the variations in the analysis window length and number of AT repetitions. As expected, robust and accuracy simulations using a fixed number of AT repetition show an identical behaviour pattern to that of the MA method. If a fixed analysis window length is used and the AT repetition is increased (from 1 to 10 repetitions) then the robustness would also increase for varying levels of noise. A narrow analysis window does not offer any significant alteration to the robustness or accuracy of the AT method for varying numbers of AT repetition. However for wide analysis windows, increasing the number of AT repetition improves its robustness and decreases its accuracy (Figure 3.16).

Variation on length of the analysis window introduces a trade-off between the robust method's robustness and accuracy. For varying levels of noise, the robustness of the AT method improves at an exponential rate as the number of AT repetition is increased linearly. This is in accordance with the original design theory of the AT method, where the vigour of a clean power spectrum is more feasible to maintain with a higher number of AT repetition when noise is introduced.

Comparing the robust LP analysis methods to the conventional autocorrelation and SEEVOC methods shows an improvement in its robustness for comparable accuracy (as shown in Figures 3.20 and 3.21). For robust simulation in high levels of noise, the AT method offers less SD, up to 3 dB lower, in comparison to the conventional LP analysis methods. The AT method, followed by the MM and MA methods, offer the most robust method of LP analysis when the speech signal is affected by high levels of noise, whilst offering no degradation in its accuracy.

6.1.2 Quantisation Performance

The quantisation of the LP parameters was simulated using split VQ in Chapter 4. A constant setting is selected for the robust LP analysis methods, where the analysis window was set to 21 frequency samples and AT repetition set to 3. This provides a good balance between maintaining the accuracy of the LPC spectrum and achieving robustness when noise is introduced to the speech signal.

The quantisation performance was measured by determining the minimum MSE between the test vector partition and the VQ codebook vector partitions. Quantisation selection criterion via the calculation of the minimum SD between the complete test vector and the VQ codebook vectors results in a more accurate overall quantisation performance (less average SD). However as the improvement in the average SD of approximately 0.01 dB is at a high cost of computational complexity, thus quantisation is performed using the minimum MSE selection criterion (Table 4.4).

In observing the quantisation performance, the robust methods have performed well for applications in scalar and vector quantisation applications. The split VQ method is applied for varying partitions and the robust method has outperformed the conventional autocorrelation and SEEVOC methods of LP analysis in terms of its SD. The AT method is shown to be the best method in terms of quantisation followed by the MM and MA methods.

For split VQ using 2, 3 and 5 parts for a 10^{th} order LP analysis, the AT method improves the average SD by approximately 0.2 dB and significantly decreases the percentage of the outlier frames (frames with SD $\geq$ 2 dB). By observing the quantisation performance of the AT method in comparison to the conventional autocorrelation method, equal level of average SD (including its outliers) may be achieved with 2-3 less quantisation bits/frame.

6.1.3 Low Bit-rate Speech Coding Application

In Chapter 5, the robust LP analysis methods were applied to the CELP coder in order to simulate the application of the robust methods in low bit-rate speech coders. A selection of real world and Gaussian noises was added to the speech signal to simulate the introduction of speech into noisy environments. Speech quality is determined in a mathematical sense by calculating the segmental-SNR and in an objective manner via listening directly to the speech.

The robust methods have been shown to produce less quantisation distortion in comparison to the conventional methods. The methods achieve spectral transparency with less bits/frame than the conventional LP analysis methods. This leads to the robust methods having a higher quality synthesised signal output.

Simulation of the MM method using narrow analysis window has been investigated to provide a higher quality speech without significantly decreasing its robustness in a noisy environment (Section 5.3). For high noise levels, where the background noise is clearly evident in speech, the robust methods have been shown to outperform both the autocorrelation and SEEVOC methods in terms of speech quality.

Listening to the synthesised speech samples, for high-levels of noise and low bit-rate of 18 bits/frame, there exists a difference in speech quality between the autocorrelation method and the robust methods. Although this is not obvious for all speech signals, speech synthesised using the robust methods are observed to be less distorted, with the actual speech observed slightly more clearly over noise.

From the simulations detailed in this dissertation, the AT method can be seen to provide the highest improvement in robustness for situations where speech is introduced into a noisy environment, followed by the MM and MA methods. In addition, the AT method achieves best performance in terms of quantisation. Experiments show that all the proposed methods offer improvements in terms of robustness and quantisation performance over the autocorrelation method of LP analysis. Less distortion in speech is also observed when the robust methods are applied to the CELP coder.

6.2 Future Work

In this dissertation, three robust LP analysis methods are proposed for analysing speech in noisy environments. These methods are found to be very effective for low bit-rate speech coding. The application of these robust methods on other areas of speech processing, such as speech recognition and speaker verification, should also be investigated, as they would benefit from these methods.

Most speech coding research in noisy conditions focuses on either the enhancement of speech, detection of pauses in speech, or noise cancellation. Future work may be concentrated in combining these robust LP analysis methods with other methods that are aimed to improve the robustness of speech coding systems, such as the *denoising* of speech using wavelets, spectral subtraction, Wiener filtering, etc.

Bibliography

[1] Atal, B. S., Cuperman, V., Gersho, A., *Speech and Audio Coding for Wireless and Network Applications*, Kluwer Academic Publishers, Boston, 1993.

[2] Atal, B. S., Cuperman, V., Gersho, A., *Advances in Speech Coding*, Kluwer Academic Publishers, Boston, 1991.

[3] Oppenheim, A. V., Schafer, R. W., *Digital Signal Processing*, Prentice Hall, Englewood Cliffs, New Jersey, 1975.

[4] Barnwell III, T. P., Nayebi, K., Richardson, C. H., *Speech Coding: A Computer Laboratory Textbook*, John Wiley and Sons Publishing, New York, 1996.

[5] Gibson, J. D., *Digital Compression for Multimedia: Principles and Standards*, Morgan Kaufmann Publishers, San Fransisco, 1998.

[6] Makhoul, J., "Linear Prediction: A Tutorial Review", in *Proc. IEEE*, Vol. 63(4), pp. 561-580, April 1975.

[7] Syrdal, A., Bennett, R., Greenspan, S., *Applied Speech Technology*, CRC Press, Boca Raton, 1995.

[8] Huang, X., Acero, A., Hon, H., *Spoken Language Processing: A Guide to Theory, Algorithms, and System Development*, Prentice Hall, New Jersey, 2001.

[9] O'Shaughnessy, D., *Speech Communication: Human and Machine*, Addison-Wesley Publishing Company, Reading, Massachusetts, 1987.

[10] Rabiner, L. R., Gold, B., *Theory and Application of Digital Signal Processing*, Prentice Hall, London, 1975.

[11] Max, J., "Quantization for Minimum Distortion", *IRE Trans. Inform. Theory*, Vol. IT-6, pp. 7-12, Mar. 1960.

[12] Paez, M. D., Glisson, T. H., "Minimum Mean-Squared-Error Quantization in Speech PCM and DPCM Systems", *IEEE Trans. Commun.*, Vol. COM-20, pp225-230, April 1972.

[13] LLoyd, S. P., "Least Squares Quantization in PCM", *IEEE Trans. Inform. Theory*, Vol. IT-28(2), pp. 129-137, Mar. 1982.

[14] Wood, R. C., "On Optimum Quantization", *IEEE Trans. Inform. Theory*, Vol. IT-15, pp. 242-252, Mar. 1969.

[15] O'Neal Jr., J. B., Stroh, R. W., "Differential PCM for Speech and Data Signals", *IEEE Trans. Commun.*, Vol. COM-20, pp. 900-912, October 1972.

[16] Oliver, B. M., Pierce, J. R., Shannon, C. E., "The Philosophy of PCM", in *Proc. IRE*, Vol. 36, pp. 1324-1331, Nov. 1948.

[17] Noll, P., "The Performance of PCM and DPCM Speech Coders in the Presence of Independent and Correlated Errors", *Conf. Rec., 1975 IEEE Int. Conf. on Commun.*, Vol. II, pp. 301-305, June 1975.

[18] Jayant, N. S., "Digital Coding of Speech Waveforms: PCM, DPCM, and DM Quantizers", in *Proc. IEEE*, Vol. 62, pp. 611-632, May 1974.

[19] O'Neal Jr., J. B., "A Bound on Signal-to-Quantizing Noise Ratios for Digital Encoding Systems", in *Proc. IEEE*, Vol. 55, pp. 287-292, Mar. 1967.

[20] Linde, Y., Buzo, A., Gray, R. M., "An Algorithm for Vector Quantizer Design", *IEEE Trans. Commun.*, Vol. COM-28(1), pp. 84-95, January 1980.

[21] Shannon, C. E., *The Mathematical Theory of Communication*, University of Illinois Press, Urbana, 1949.

[22] Davisson, L. D., "Rate-Distortion Theory and Application", in *Proc. IEEE*, Vol. 60, pp. 800-808, July 1972.

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

[24] Shoham, Y., "Cascaded Likelihood Vector Coding of the LPC Information", in *Proc. ICASSP*, 1989, Vol. 1, pp. 160-163.

[25] Gray, R. M., "Vector Quantization", in *IEEE ASSP Magazine*, pp. 4-29, April 1984.

[26] Sayood, K., *Introduction to Data Compression*, Morgan Kaufmann Publishing, San Fransisco, 1996.

[27] Atal, B. S., Remde, J. R., "A New Model of LPC Excitation for Producing Natural-Sounding Speech at Low Bit Rates", in *Proc. ICASSP*, 1982, pp. 614-617.

[28] Atal, B. S., "High Quality Speech at Low Bit Rates: Multipulse and Stochastically Excited Linear Predictive Coders", in *Proc. ICASSP*, 1986, pp. 1681-1684.

[29] Singhal, S., Atal, B. S., "Amplitude Optimization and Pitch Prediction in Multipulse Coders", *IEEE Trans. Acoust., Speech, Signal Processing*, Vol. 37, No. 3, pp. 317-327, Mar. 1989.

[30] Berouti, M., Garten, H., Kabal, P., Mermelstein, P., "Efficient Computation and Encoding of the Multipulse Excitation for LPC", in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Processing*, 1984, pp. 384-387.

[31] Rabiner, L. R., Schafer, R. W., *Digital Processing of Speech Signals*, Prentice Hall, London, 1978.

[32] Schroeder, M., Atal, B., "Code Excited Linear Prediction: High Quality Speech at Very Low Bit Rates", in *Proc. ICASSP*, 1985, Vol. 3, pp. 937-940.

[33] Atal, B. S., Schroeder, M. R., “Predictive Coding of Speech Signals and Subjective Error Criteria”, *Bell Systems Technology Journal*, Vol. 49, pp. 1973-1986, October 1970.

[34] Atal, B. S., Hanauer, S. L., “Speech Analysis and Synthesis by Linear Prediction of Speech Wave”, *J. Acoust. Soc. Am.*, Vol. 40, pp. 637-655, Aug. 1971.

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

[36] Owens, F. J., *Signal Processing of Speech*, MacMillan series, Basingstoke, 1993.

[37] Jayant, N. S., Noll, P., *Digital Coding of Waveforms: Principles and Applications to Speech and Video*, Prentice Hall, Englewood Cliffs, New Jersey, 1984.

[38] Gold, B., Morgan, N., *Speech and Audio Signal Processing: Processing and Perception of Speech and Music*, John Wiley and Sons Publishing, New York, 2000.

[39] Strobach, P., *Linear Prediction Theory: A Mathematical Basis for Adaptive Systems*, Springer-Verlag, Berlin, 1990.

[40] Schroeder, M. R., “Linear Prediction, Extremal Entropy and Prior Information in Speech Signal Analysis and Synthesis”, *Speech Commun.*, Vol. 1, No. 1, pp. 9-20, 1982.

[41] Kay, S. M., *Modern Spectral Estimation: Theory and Application*, Prentice Hall, Englewood Cliffs, New Jersey, 1988.

[42] Stoica, P., Moses, R. L., *Introduction to Spectral Analysis*, Prentice Hall, New Jersey, 1997.

[43] Priestley, M. B., *Spectral Analysis and Time Series*, Academic Press, London, 1981.

[44] Marple Jr., S. L., *Digital Spectral Analysis: with Applications*, Prentice Hall, Englewood Cliffs, New Jersey, 1987.

[45] Pillai, S. U., *Spectrum Estimation and System Identification*, Springer-Verlag, New York, 1993.

[46] Broersen, P. M. T., "Facts and Fiction in Spectral Analysis", *IEEE Trans. Instrum. Meas.*, Vol. 49, No. 4, Aug. 2000.

[47] Martin, R. J., "Autoregression and Irregular Sampling: Spectral Estimation", *Signal Processing Journal*, Vol. 77, No. 2, pp. 139-157, Sep. 1999.

[48] Conforto, S., D'Alessio, T., "Optimal Estimation of Power Spectral Density by Means of a Time-varying Autoregressive Approach", *Signal Processing Journal*, Vol. 72, No. 1, pp. 1-14, Jan. 1999.

[49] Geckinli, N. C., *Discrete Fourier Transformation and its Applications to Power Spectra Estimation*, Elsevier, Amsterdam, 1983.

[50] Proakis, J. G., Manolakis, D. G., *Digital Signal Processing: Principles, Algorithms, and Applications*, 3rd ed, Prentice Hall, New Jersey, 1996.

[51] Deller Jr., J. R., Hansen, J. H. L., Proakis, J. G., *Discrete-time Processing of Speech Signals*, Institute of Electrical and Electronics Engineers, New York, 2000.

[52] Kondoz, A. M., *Digital Speech: Coding for Low Bit Rate Communication Systems*, John Wiley and Sons Publishing, England, 1994.

[53] Rayner, J. N., *An Introduction to Spectral Analysis*, Pion Ltd., London, 1971.

[54] Kay, S. M., Marple Jr., S. L., "Spectrum Analysis: A Modern Perspective", in *Proc. IEEE*, Vol. 69(11), pp. 1380-1419, Nov. 1981.

[55] Atal, B. S., "Linear Prediction Analysis of Speech Based on Pole-zero Representation", *J. Acoust. Soc. Am.*, Vol. 64(5), pp. 1310-1318, 1978.

[56] Atal, B. S., Schroeder, M. R., “Linear Prediction Analysis of Speech Based on a Pole-Zero Representation”, *J. Acoust. Soc. Am.*, Vol. 64(5), pp. 1310-1318, Nov. 1978.

[57] Dembo, A., Zeitouni, O., “Maximum A Posteriori Estimation of Time-Varying ARMA Processes from Noisy Observations”, *IEEE Trans. Acoust., Speech, Signal Processing*, Vol. 36(4), pp. 471-476, 1988.

[58] Cohen, I., Berdugo, B., “Noise Estimation by Minima Controlled Recursive Averaging for Robust Speech Enhancement”, *IEEE Signal Processing Lett.*, Vol. 9, No. 1, pp. 12-15, Jan. 2002.

[59] Zhao, Y., “Spectrum Estimation of Short-Time Stationary Signals in Additive Noise and Channel Distortion”, *IEEE Trans. Signal Processing*, Vol. 49, No. 7, pp. 1409-1420, July 2001.

[60] Sasou, A., Tanaka, K., “Robust LP Analysis Using Glottal Source HMM with Application to High-Pitched and Noise Corrupted Signal”, in *Proc. 7th European Conf. Speech, Commun., Technology*, Aalborg, Denmark, Sep. 2001, pp. 2443-2446.

[61] Paul, D. B., “The Spectral Envelope Estimation Vocoder”, *IEEE Trans. Acoust., Speech, Signal Processing*, ASSP-29, pp. 786-794, Aug. 1981.

[62] Zhang, W., Holmes, W. H., “Performance And Optimization Of The SEEVOC Algorithm”, in *Proc. 5th Int. Conf. Spoken Language*, 1998, Vol. 2, pp. 523-526.

[63] Schwarz, H. R., *Numerical Analysis: A Comprehensive Introduction*, John Wiley and Sons Publishing, Chichester, 1989.

[64] Atal, B., “Predictive Coding of Speech at Low Bit Rates”, *IEEE Trans. Commun.*, COM-30(4), pp. 600-614, May 1982.

[65] Ramachandran, R. P., Mammone, R. J., *Modern Methods of Speech Processing*, Kluwer Academic Publishers, Boston, 1995.

[66] Kroon, P., Atal, B. S., "Quantization Procedures for the Excitation in CELP Coders", in *Proc. ICASSP*, 1987, pp. 1649-1652.

[67] Krishna, K., Murty, V. L. N., Ramakrishnan, V. L. N., "Vector Quantization of Excitation Gains in Speech Coding", *Signal Processing Journal*, Vol. 81, No. 1, pp. 203-209, Jan. 2001.

[68] Kleijn, W. B., Paliwal, K. K., *Speech Coding and Synthesis*, Elsevier, Amsterdam, 1995.

[69] Wakita, H., "Linear Prediction Voice Synthesizers: Line Spectrum Pairs (LSP) is the Newest of Several Techniques", in *Speech Tecnhnology*, pp. 17-22, Fall 1981.

[70] Gray, A. H., Markel, J. D., "Quantization and Bit Allocation in Speech Processing", *IEEE Trans. Acoust., Speech, Signal Processing*, Vol. ASSP-24, No. 6, pp. 459-473, June 1976.

[71] Viswanathan, R., Makhoul, J., "Quantization Properties of Transmission Parameters in Linear Predictive Systems", *IEEE Trans. Acoust., Speech, Signal Processing*, Vol. ASSP-23, No. 3, pp. 309-321, June 1975.

[72] Itakura, F., "Line Spectrum Representation of Linear Predictive Coefficients of Speech Signals", *J. Acoust. Soc. Am.*, Vol. 57, p. S35, 1975.

[73] Soong, F., Juang, B., "Line Spectrum Pair (LSP) and Speech Data Compression", in *Proc. ICASSP*, 1984, pp. 1.10.1-1.10.4.

[74] Grassi, S., Dufaux, A., Ansorge, M., Pellandini, F., "Efficient Algorithm to Compute LSP Parameters from 10th-order LPC Coefficients", in *Proc. ICASSP*, 1997, Vol. 3, pp. 1707-1710.

[75] Tourneret, J. Y., "Statistical Properties of Line Spectrum Pairs", *Signal Processing Journal*, Vol. 65, No. 2, pp. 239-255, 1998.

[76] Kovesi, B., Saoudi, S., Boucher, J. M., Horvath, G., "Real Time Vector Quantization of LSP Parameters", *Speech Commun.*, Vol. 29, No. 1, pp. 39-47, Sep. 1999.

[77] Sugamura, N., Itakura, F., "Speech Analysis and Synthesis Methods Developed at ECL in NTT (from LPC to LSP)", *Speech Commun.*, Vol. 5, pp. 199-215, 1986.

[78] Cohn, R. P., Collura, J. S., "Incorporating Perception into LSF Quantization - Some Experiments", in *Proc. ICASSP*, 1997, Vol. 2, pp. 1347-1351.

[79] Nein, H. W., Lin, C. T., "Incorporating Error Shaping Technique into LSF Vector Quantization", *IEEE Trans. Acoust., Speech, Signal Processing*, Vol. 9, No. 2, pp. 73-86, 2001.

[80] Noll, P., "Adaptive Quantizing in Speech Coding Systems", in *Proc. 1974 IEEE Zurich Seminar on Digital Commun.*, pp. B3(1)-B3(6), March 1974.

[81] Jayant, N. S., *Waveform Quantization and Coding*, IEEE Press, New York, 1976.

[82] Paliwal, K. K., "Robust Linear Prediction Analysis Methods and Their Application to Speech Coding and Recognition", paper under preparation.

[83] Paliwal, K. K., Koestoer, N. P., "Robust Linear Prediction Analysis for Low Bit-Rate Speech Coding", in *Proc. Fourth Australasian Workshop on Signal Processing Applications*, Brisbane, Australia, Dec. 2002.

[84] Paliwal, K. K., Atal, B. S., "Efficient Vector Quantization of LPC Parameters at 24 Bits/Frame", *IEEE Trans. Speech Audio Processing*, Vol. 1, No. 1, Jan. 1993.

[85] Makhoul, J., Roucos, S., Gish, H., "Vector Quantization in Speech Coding", in *Proc. IEEE*, Vol. 73, No. 11, pp. 1551-1588, November 1985.

[86] Gonzalez, R. C., Wood, R. E., *Digital Image Processing*, Addison-Wesley Publishing Company, Reading, Massachusetts, 1992.

[87] Ackenhausen, J., *Signal Processing Technology and Applications*, IEEE Press, Piscataway, New Jersey, 1995.

[88] Farvardin, N., Modestino, J. W., "Optimum Quantizer Performance for a Class of Non-Gaussian Memoryless Sources", *IEEE Trans. Inform. Theory,* Vol. IT-30(3), pp. 485-497, May 1984.

[89] Miyoshi, Y., Yamata, K., Mizoguchi, R., Yanagida, M., Kakusho, O., "Analysis of Speech Signals of Short Pitch Period by a Sample Selective Linear Prediction", *IEEE Trans. Acoust., Speech, Signal Processing*, Vol. 35, No. 9, pp. 1223-1240, 1987.

[90] Cappe, O., Oudot, M., Moulines, E., "Spectral Envelope Estimation using a Penalized Likelihood Criterion", in *IEEE ASSP Workshop*, Mohonk, USA, Oct. 1997.

[91] Molyneux, D. J., Parris, C. I., Sun, X. Q., Cheetham, B. M. G., "Comparison of Spectral Estimation Techniques for Low Bit-Rate Speech Coding", *IEEE ICSLP 1998*, Sydney, Dec. 1998.

[92] Campedel-Oudot, M., Cappe, O., Moulines, E., "Estimation of the Spectral Envelope of Voiced Sounds Using a Penalized Likelihood Approach", *IEEE Trans. Speech Audio Processing*, Vol. 9, No. 5, July 2001.

[93] Harborg, E., Knudsen, J., Fuldseth, A., Johansen, F., "A real time wideband CELP coder for a videophone application", in *Proc. ICASSP*, Apr. 1994, Vol. 2, pp. 121-124.

[94] Zhang, L., Wang, T., Cuperman, V., "A CELP Variable Rate Speech Codec with Low Average Rate", in *Proc. ICASSP*, Apr. 1997, Vol. 2, pp. 735-738.

[95] Hayashi, S., Suguimoto, M., "Low Bit-rate CELP Speech Coder with Low Delay", *Signal Processing Journal*, 1999, Vol. 72, No. 2, pp. 97-105.

[96] Choi, S. H., Kim, H. K., Lee, H. S., “Speech Recognition Using Quantized LSP Parameters and Their Transformations in Digital Communications”, *Speech Commun.*, Vol. 30, No. 4, pp. 223-233, Apr. 2000.

[97] Zhu, Q., Iseli, M., Cui, X., Alwan, A., “Noise Robust Feature Extraction for ASR using the Aurora 2 Database”, in *Proc. 7th European Conf. Speech, Commun., Technology*, Aalborg, Denmark, Sep. 2001, pp. 185-188.

[98] Chen, J., Paliwal, K. K., Nakamura, S., “Sub-Band Based Additive Noise Removal for Robust Speech Recognition”, in *Proc. 7th European Conf. Speech, Commun., Technology*, Aalborg, Denmark, Sep. 2001, pp. 571-574.

[99] Attias, H., Deng, L., Acero, A., Platt, J. C., “A New Method fpr Speech Denoising and Robust Speech Recognition Using Probabilistic Models for Clean Speech and for Noise”, in *Proc. 7th European Conf. Speech, Commun., Technology*, Aalborg, Denmark, Sep. 2001, pp. 1903-1906.

[100] Stachurski, J., “A Pitch Pulse Evolution Model for Linear Predictive Coding of Speech”, *PhD. Thesis*, Department of Electrical Engineering, McGill University, Montreal, Canada, 1998.

[101] *Webster’s New Dictionary and Thesaurus*, Geddes & Grosset Ltd., New Lanark, Scotland, 1990.

[102] *Chambers Science and Technology Dictionary*, Chambers Ltd., Edinburgh, Scotland, 1991.

[103] Press, W. H., *Numerical Recipes in C: The Art of Scientific Computing*, Cambridge University Press, Cambridge, New York, 1992.