INFORMATION TO USERS

This manuscript has been reproduced from the microfilm master. UMI films the text directly from the original or copy submitted. Thus, some thesis and dissertation copies are in typewriter face, while others may be from any type of computer printer.

The quality of this reproduction is dependent upon the quality of the copy submitted. Broken or indistinct print, colored or poor quality illustrations and photographs, print bleedthrough, substandard margins, and improper alignment can adversely affect reproduction.

In the unlikely event that the author did not send UMI a complete manuscript and there are missing pages, these will be noted. Also, if unauthorized copyright material had to be removed, a note will indicate the deletion.

Oversize materials (e.g., maps, drawings, charts) are reproduced by sectioning the original, beginning at the upper left-hand corner and continuing from left to right in equal sections with small overlaps.

ProQuest Information and Learning
300 North Zeeb Road, Ann Arbor, MI 48106-1346 USA
800-521-0600

Coding of Excitation Signals In a Waveform Interpolation Speech Coder

Mohammad M. A. Khan

Department of Electrical & Computer Engineering
McGill University
Montreal, Canada

July 2001

A thesis submitted to the Faculty of Graduate Studies and Research in partial fulfillment of the requirements for the degree of Master of Engineering.

Abstract

The goal of this thesis is to improve the quality of the Waveform Interpolation (WI) coded speech at 4.25 kbps. The quality improvement is focused on the efficient coding scheme of voiced speech segments, while keeping the basic coding format intact. In the WI paradigm voiced speech is modelled as a concatenation of the Slowly Evolving pitch-cycle Waveforms (SEW). Vector quantization is the optimal approach to encode the SEW magnitude at low bit rates, but its complexity imposes a formidable barrier.

Product code vector quantizers (PC-VQ) are a family of structured VQs that circumvent the complexity obstacle. The performance of product code VQs can be traded off against their storage and encoding complexity. This thesis introduces split/shape-gain VQ—a hybrid product code VQ, as an approach to quantize the SEW magnitude. The amplitude spectrum of the SEW is split into three non-overlapping subbands. The gains of the three subbands form the gain vector which are quantized using the conventional Generalized Lloyd Algorithm (GLA). Each shape vector obtained by normalizing each subband by its corresponding coded gain is quantized using a dimension conversion VQ along with a perceptually based bit allocation strategy and a perceptually weighted distortion measure. At the receiver, the discontinuity of the gain contour at the boundary of subbands introduces buzziness in the reconstructed speech. This problem is tackled by smoothing the gain versus frequency contour using a piecewise monotonic cubic interpolant. Simulation results indicate that the new method improves speech quality significantly.

The necessity of SEW phase information in the WI coder is also investigated in this thesis. Informal subjective test results demonstrate that transmission of SEW magnitude encoded by split/shape-gain VQ and inclusion of a fixed phase spectrum drawn from a voiced segment of a high-pitched male speaker obviates the need to send phase information.

Sommaire

Le but de cette thèse est d'améliorer la qualité de la parole codée à 4.25 kbps avec un codeur basé sur l'interpolation du signal (abrégé WI pour *waveform interpolation*). L'amélioration de la qualité est concentrée sur le codage efficace de segments de la parole, sans toutefois modifier le format de base du codeur. Dans le paradigme WI, la parole voisée est représentée par l'enchaînement des SEW (*Slowly Evolving pitch-cycle Waveforms*). La quantification vectorielle (VQ) est l'approche optimale pour le codage des SEW à bas débit binaire, mais elle pose des problèmes de complexité.

Les quantificateurs PC-VQ (*product code - vector quantization*) sont un sous-groupe des VQ structurés qui évitent cet obstacle de complexité. La performance des PC-VQ peut être compensée aux dépends de leur complexité. Cette thèse introduit *split/shape-gain* VQ, un PC-VQ hybride, pour la quantification de la grandeur des SEW. Le spectre des SEW est divisé en trois bandes sans superposition. Les gains de chaque bande forment ensemble les vecteurs de gains, qui sont quantifiés en utilisant le *Generalized Lloyd Algorithm* (GLA). Chaque vecteur, obtenu en normalisant chaque bande par leur gain codé, est quantifié en utilisant une conversion de dimension VQ avec une stratégie d'allocation de bits basée sur la perception et une mesure de distorsion perceptuelle. Au récepteur, la distorsion des gains aux bornes des sous-bandes introduit des bourdonnements dans la parole reconstruite. Une nouvelle manière de lisser les gains en fréquence est présentée pour résoudre ce problème. Les résultats obtenus par simulation indiquent que la nouvelle méthode améliore la qualité du signal de parole de façon significative.

La nécessité de l'information contenue dans la phase des SEW est également étudiée dans cette thèse. Des examens subjectifs informels démontrent que la transmission de la grandeur des SEW codée avec *split/shape-gain* VQ, combinée avec une phase fixe tirée d'un segment de parole voisée, élimine le besoin de transmettre l'information contenue dans la phase.

Acknowledgments

I would like to thank my supervisor Prof. Peter Kabal for his guidance throughout my graduate studies. This work would have not been accomplished without his valuable advice and his patience.

The feedback and motivation I received from Dr. Hossein Najafzadeh-Azghandi and Yasheng Qian also proved to be helpful. Moreover, I thank Christopher Cave and Benoît Pelletier for the French translation of the thesis abstract.

Portions of manuscript were proofread by Aziz Shallwani, Alexander Wyglinski, Tarun Agarwal and Mark Klein. Their suggestions improved the quality of certain chapters.

My gratitude goes to the International Council for Canadian Studies for awarding me the Commonwealth Scholarship.

I sincerely thank all my friends in the Telecommunications and Signal Processing laboratory, past and present, for many interesting discussions. Special thanks to Wesley Pereira and Ejaz Mahfuz for their support and companionship.

Finally, my deepest gratitude goes to my parents, my brothers and my sister for their constant encouragement in this very rewarding effort.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Motivation for Low Bit-Rate Speech Coding

Speech is an acoustic waveform that conveys information from a speaker to a listener. When two parties are at a distance from each other there must be a medium to transmit the speech signals. There are two types of transmission: analog transmission and digital transmission. Uncompressed digital speech consumes a large amount of storage space and transmission bandwidth. Compressed digital transmission of speech is more versatile than analog, providing the opportunity of achieving lower costs, consistent quality and security.

Digital speech coding or speech compression is concerned with obtaining compact digital representations of voice signals. The objective in speech coding is to represent speech with a small number of bits while maintaining its perceptual quality. Perceptually irrelevant information in the speech signal makes it possible to encode speech at low bit-rates.

The capability of speech compression has been central to the technologies of robust long-distance communications and high-quality speech storage. Compression continues to be a key technology in communications in spite of the promise of optical transmission media of relatively unlimited bandwidth. This is because of our continued and, in fact, increasing need to use band-limited media such as radio and satellite links. Furthermore, storage and archival of large volumes of spoken information makes speech compression essential even in the context of significant increases in the capacity of storage.

Since the bandwidth of a signal is a function of its bit-rate, low bit-rate speech technology is a key factor in meeting the increasing demand for new digital wireless communication services. Impressive progress has been made during recent years in coding speech with high

quality at low bit-rates and at low cost. The rapid advancement in the efficiency of digital signal processors and digital signal processing techniques have stimulated the development of speech coding algorithms. These trends entail a continued interest in speech compression technology as they provide a viable means to realize reduced operating costs in voice communication systems.

1.2 Speech: Production, Properties and Perception

Our hearing system isn't equally sensitive to distortions at different frequencies and has a limited dynamic range. So, the understanding of the physiology of human speech production, the basic properties of the speech signal and its perception is crucial to the design of a speech coder which would, ideally, parameterize only perceptually relevant information and thus compactly represent the signal.

Speech Production

Fig. 1.1 portrays a medial saggital section of the human speech production system. As the

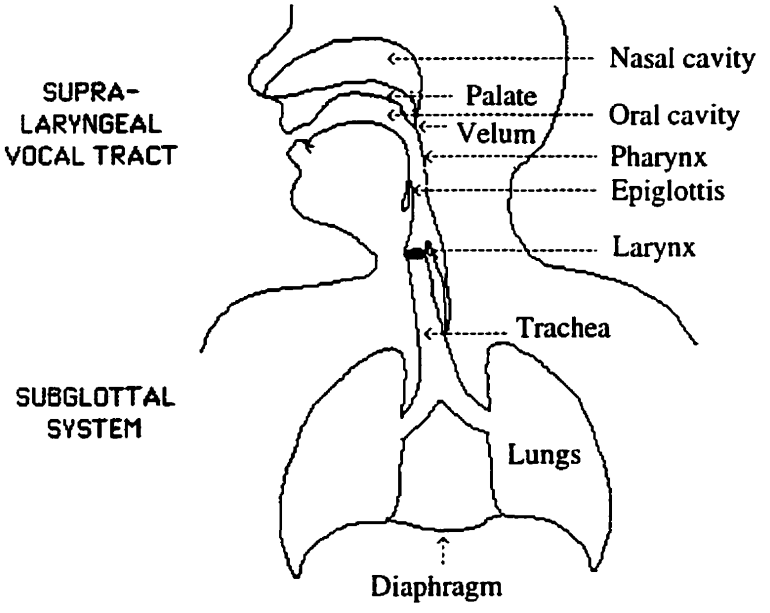


Fig. 1.1 Schematic view of human speech production mechanism. (From [1].)

diaphragm forces air through the system, the voluntary movements of anatomical structures of this system generate and shape a wide variety of waveforms. These waveforms can be broadly categorized into voiced and unvoiced speech.

With *voiced* speech, air pressure from the lungs forces normally closed vocal folds (or cords)[1] to open and vibrate in a relaxation oscillation. The frequency of this vibration is known as the *pitch frequency*, $F0$, and it varies from 50 to 400 Hz depending on the shape and tension in the vocal cords, and the pressure of the air behind them. The effect of this opening and closing of the glottis (the space between the vocal cords) is that the air passing through the rest of the vocal tract appears as a quasi-periodic pulse train.

Unvoiced sounds result when the excitation is a noise-like turbulence produced by forcing air at high velocities through a constriction in the vocal tract (pharyngeal cavity + oral cavity) while the glottis is held open.

At this point the speech signal consists of a series of pulses or random noise depending on whether the speech is voiced or unvoiced. As they propagate through the rest of the vocal tract, their frequency spectrum is shaped by the frequency response of the vocal tract. Thus, the vocal tract acts as a *spectral shaping filter* and its frequency selectivity is governed by its size and shape.

A class of sounds called *plosives* results when a complete closure is made in the vocal tract, and air pressure is built up behind this closure and released suddenly.

Properties of Speech

Different speech sounds are formed by varying the shape of the vocal tract. Thus, the spectral properties of the speech signal vary with time as the vocal tract shape varies. However, speech can be considered as quasi-stationary over short segments, typically 5–20 ms. The statistical and spectral properties of speech are thus defined over short segments.

Different speech sounds are distinguished by the human ear on the basis of their short time spectra and how these spectra evolve with time. The effective bandwidth of speech is approximately 7 kHz.

In speech production, especially for *voiced sounds* like vowels, the vocal tract acts as a resonant cavity. For most people, the resonance frequencies are centered at 500 Hz and its odd harmonics. This resonance produces large peaks in the resulting spectrum of vowels.

[1]The *vocal folds* are two membranes situated in the larynx. These membranes allow the area of the trachea (the glottis) to be varied.

These peaks are known as formants. The first three formants, usually occurring below 3 kHz, are quite important in speech perception. The time-domain voiced signals are quasi-periodic due to repeated excitations of the vocal tract by vocal fold closures. Thus, voiced speech has *line spectra* with frequency spacing of $F0$ Hz. An example of voiced speech is given in Fig. 1.2. The periodic nature of the speech is clearly visible. The first four

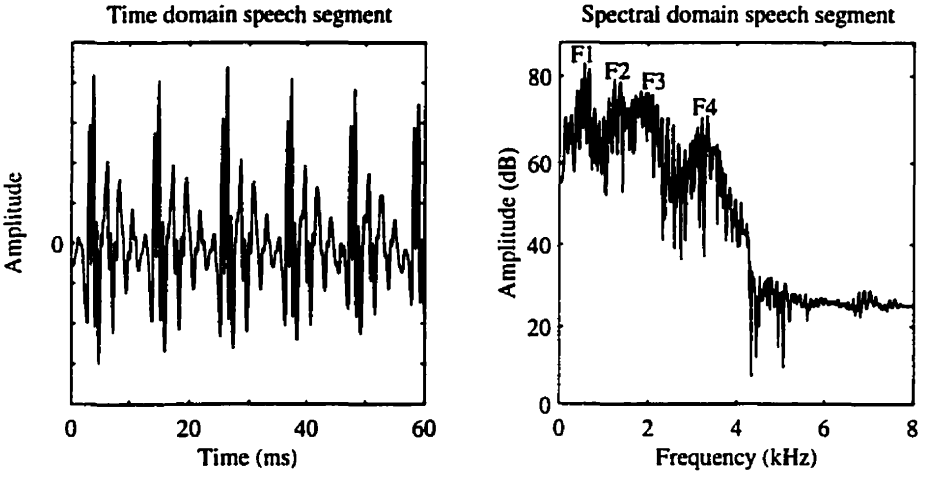


Fig. 1.2 Voiced speech: the time signal and its spectrum.

formants are labelled in the spectrum. The fine harmonic structure (narrow peaks), due to the vibrating vocal folds, is also visible in the spectrum. It should be noted that the formant structure (spectral envelope) appears to break down above 4 kHz as noise introduced by the turbulent flow of air through the vocal tract begins to dominate. The envelope falls off at about -6 dB/octave due to the radiation from the lips and the nature of the glottal excitation [2]. The spectrum also shows the enormous dynamic range and the lowpass nature of voiced speech.

An example of *unvoiced* hiss-like sounds like /f/, /s/, /sh/ is given in Fig. 1.3. Note that the time domain samples are noise-like and aperiodic while the frequency view does not display the clear resonant peaks that are found in voiced sounds. Unvoiced speech tends to have a nearly flat or a high-pass spectrum. The energy in the signal is typically much lower than that in voiced speech.

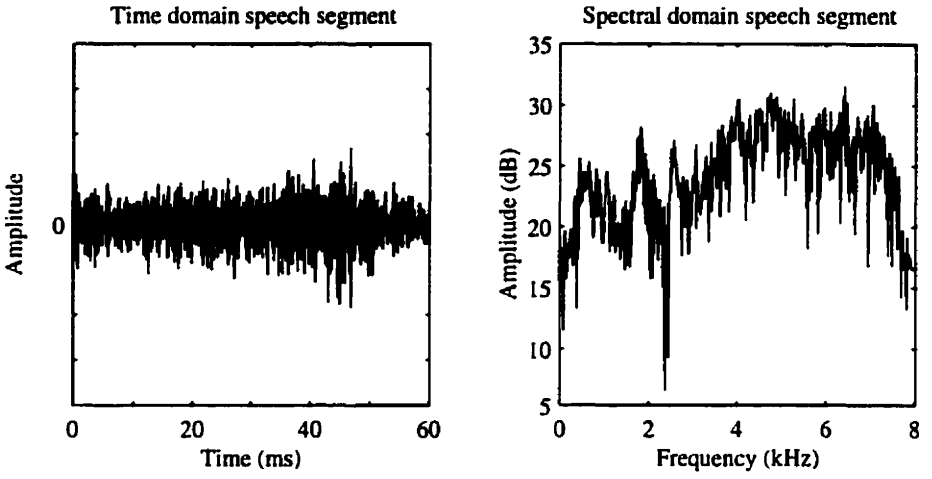


Fig. 1.3 Unvoiced speech: the time signal and its spectrum.

Speech Perception

Studying the hearing process has become a large part of present day digital audio technology. Little is known about how the brain decodes the acoustic information it receives. However, quite a lot is known about how the ear processes sounds. A cross-sectional view of the ear is shown in Fig. 1.4. The human ear is composed of three main sections: the outer ear, the middle ear, and the inner ear. The functional description of each section is given below:

Outer Ear

The outer ear has two parts; the *pinna* (*auricle*) and the *external auditory canal.* The pinna collects sounds and aids into sound localization. The external auditory canal channels the sound into the middle ear. The canal is closed at one end by the eardrum. It can be viewed as an acoustic tube that resonates at 3 kHz. This resonance amplifies energy in the 3–5 kHz range , which likely aids perception of sounds (e.g., obstruents) having significant information at these high frequencies.

Middle Ear

The middle ear begins at the *eardrum* (a thin membrane) and contains a set of three bones named *malleus* or *hammer*, *incus* or *anvil*, and *stapes* or *stirrup*. These three bones act as

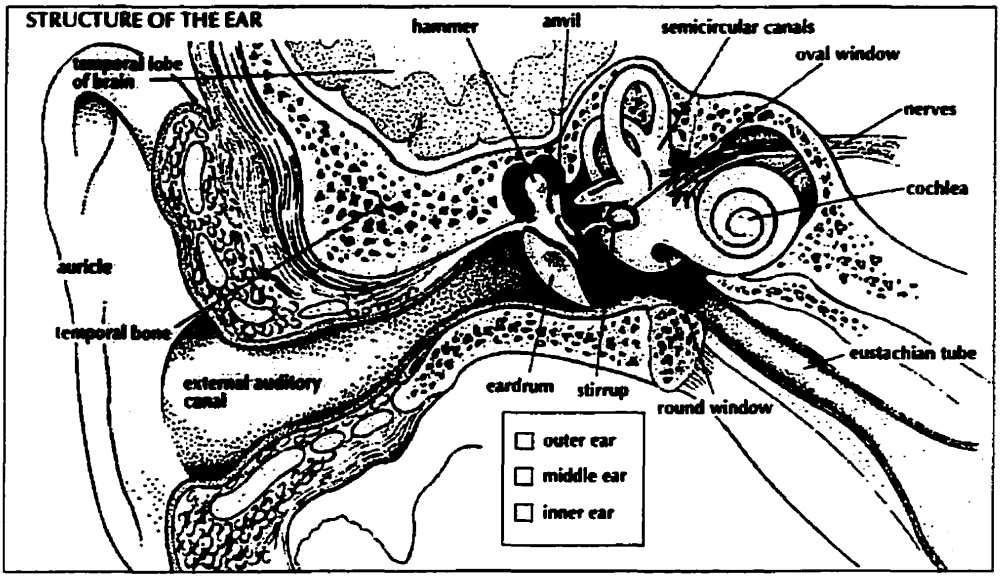


Fig. 1.4 The structure of the peripheral auditory system. (From [3].)

a transformer and match the acoustic impedance of the inner ear with that of air. Muscles attached to these bones suppress any violent vibration that may come from eardrum and protects the inner ear. This protection only works for sounds below 2 kHz and it does not work for impulsive sounds. The eustachian tube connects the middle ear to the vocal tract and removes any static pressure difference between the middle ear and the outer ear.

Inner Ear

The inner ear consists of the semicircular canals, the cochlea, and auditory nerve terminations. The semicircular canals help balancing the body and have no apparent role in hearing process. The cochlea is fluid filled and helical in shape. The cochlea is sealed by the oval and round windows. Inside the cochlea there is a hair-lined membrane called the Basilar Membrane (BM).

Properties of Ear for Speech Perception

The basilar membrane performs a crucial part of speech perception. The BM converts the mechanical signal into a neural signal. Different frequencies excite different portions of this membrane allowing a frequency analysis of the signal to be carried out. The ear is

essentially a *spectrum analyzer* that responds to the magnitude of the signal. The frequency resolution is greatest at low frequencies. The frequency resolution of the auditory system in terms of auditory filters and critical bands will be discussed in Section 4.2.1.

There is a limit to the sensitivity of the ear. If sounds are too weak they will not be detected. This is known as the *threshold of audibility*. This threshold varies with frequency and it can be increased at any given frequency by the presence of a large signal at a nearby lower frequency. This phenomenon is called *masking* and it is widely used in speech coding. If the quantization noise can be concentrated around the formants then it will be rendered inaudible to the listener.

1.3 Overview of Speech Coding

Over the past few decades, a variety of speech coding techniques has been proposed, analyzed, and developed. Here we briefly discuss those techniques which are used today, and those which may be used in the future. Traditionally, speech coders are divided into two classes—waveform coders and source coders (also known as parametric coders or vocoders). Typically waveform coders operate at high bit-rates, and give very good quality speech. Source coders are used at very low bit-rates, but tend to produce synthetic quality speech. Recently, a new class of coders, called hybrid coders, is introduced which uses techniques from both source and waveform coding, and gives good quality speech at intermediate bit-rates. Fig. 1.5 shows the typical behavior of the speech quality versus bit-rate curve for the two main classes of speech coders.

1.3.1 Waveform Coders

Waveform coders attempt to reproduce the input signal waveform. They are generally designed to be signal independent so they can be used to code a wide variety of signals. Generally they are low complexity coders which produce high quality speech at rates above about 16 kbps. Waveform coding can be carried out in either the time or the frequency domains.

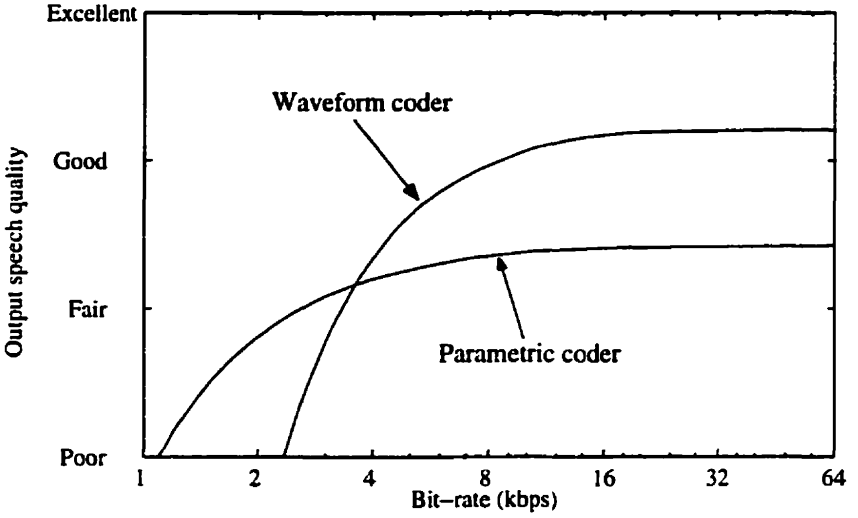


Fig. 1.5 Speech quality versus bit-rate for common classes of coders. (From [4].)

Time Domain Coders

Time domain coders perform the coding process on the time samples of the signal data. The well known coding methods in the time domain are [5]: Pulse Code Modulation (PCM), Adaptive Pulse Code Modulation (APCM), Differential Pulse Code Modulation (DPCM), Adaptive Differential Pulse Code Modulation (ADPCM), Delta Modulation (DM), Adaptive Delta Modulation (ADM), and Adaptive Predictive Coding (APC). In the following, we briefly describe some important coding schemes in the time domain.

PCM Coders

Pulse code modulation is the simplest type of waveform coding. It is essentially just a sample-by-sample quantization process. Any form of scalar quantization can be used with this scheme, but the most common form of quantization used is logarithmic quantization. The International Telegraph and Telephone Consultative Committee's (CCITT) Recommendation G.711 defines 8 bit A-law and μ-law PCM as the standard method of coding telephone speech.

DPCM and ADPCM Coders

PCM makes no assumptions about the nature of the waveform to be coded, hence it works well for non-speech signals. However, when coding speech there is a very high correlation between adjacent samples. This correlation could be used to reduce the resulting bit-rate. One simple method of doing this is to transmit only the differences between each sample. This difference signal will have a much lower dynamic range than the original speech, so it can be effectively quantized using a quantizer with fewer reconstruction levels. In the above method, the previous sample is being used to predict the value of the present sample. The prediction would be improved if a larger block of the speech is used to make the prediction. This technique is known as differential pulse code modulation (DPCM). Its structure is shown in Fig. 1.6.

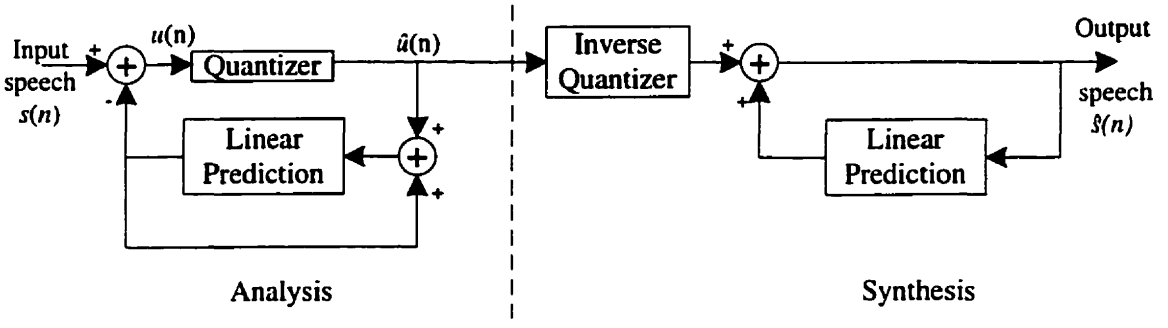


Fig. 1.6 General differential PCM system: coder on the left, decoder on the right. The inverse quantizer simply converts transmitted codes back into a single $\hat{u}(n)$ value.

An enhanced version of DPCM is Adaptive DPCM in which the predictor and quantizer are adapted to local characteristics of the input signal. There are a number of ITU recommendations based on ADPCM algorithms for narrowband (8 kHz sampling rate) speech and audio coding e.g., G.726 operating at 40, 32, 24 and 16 kbps. The complexity of ADPCM coders is fairly low.

Frequency Domain Coders

Frequency domain waveform coders split the signal into a number of separate frequency components and encode these separately. The number of bits used to code each frequency component can be varied dynamically. Frequency domain coders are divided into two

groups: subband coders and transform coders .

Subband Coders

Subband coders employ a few bandpass filters (i.e., a filterbank) to split the input signal into a number of bandpass signals (subband signals) which are coded separately. At the receiver the subband signals are decoded and summed up to reconstruct the output signal. The main advantage of subband coding is that the quantization noise produced in one band is confined to that band. The ITU has a standard on subband coding (i.e., G.722 audio coder [6]) which encodes wideband audio signals (7 kHz bandwidth sampled at 16 kHz) for transmission at 48, 56, or 64 kbps.

Transform Coders

This technique involves a block transformation of a windowed segment of the input signal into the frequency, or some other similar, domain. Adaptive Coding is then accomplished by assigning more bits to the more important transform coefficients. At the receiver the decoder carries out the inverse transform to obtain the reconstructed signal. Several transforms like the Discrete Fourier transform (DFT) or the Discrete Cosine Transform (DCT) can be used.

1.3.2 Source Coders

Source coders operate using a model of how the source was generated, and attempt to extract, from the signal being coded, the parameters of the model. It is these model parameters which are transmitted to the decoder. Source coders for speech are called vocoders, and use the source-filter model of speech production as shown in Fig. 1.7. This model assumes that speech is produced by exciting a linear time-varying filter (the vocal tract) by white noise for unvoiced speech segments, or a train of pulses for voiced speech. Vocoders operate at around 2 kbps or below and deliver synthetic quality.

Depending upon the methods of extracting the model parameters, several different types of vocoders have been developed, viz, channel vocoder, homomorphic vocoder, formant vocoder, linear prediction vocoder.

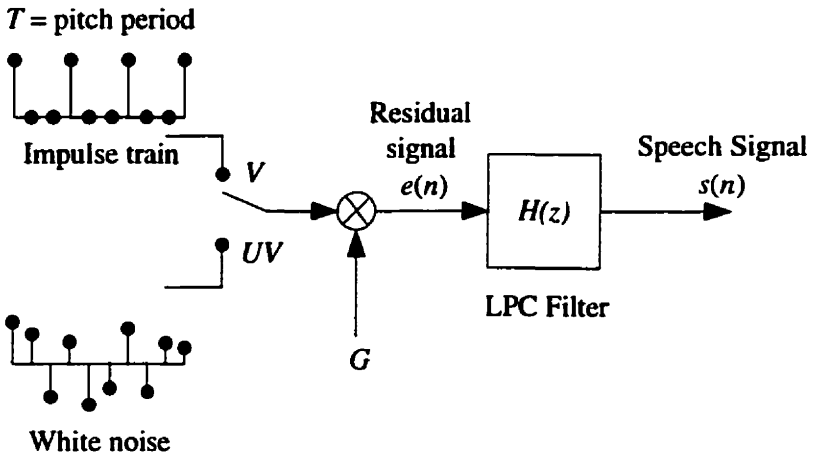


Fig. 1.7 The source-filter model of speech production used by vocoders.

1.3.3 Hybrid Coders

Hybrid coders attempt to fill the gap between waveform and parametric coders. Waveform coders are capable of providing good quality speech at bit-rates around 16 kbps; on the other hand, vocoders operate at very low bit-rates (2.4 kbps and below) but cannot provide natural quality. Although other forms of hybrid coders exist, the most successful and commonly used are time domain Analysis-by-Synthesis (AbS) coders. Such coders use the same linear prediction filter model of the vocal tract as found in LPC vocoders. However, instead of applying a simple two-state voiced/unvoiced model to find the necessary input to this filter, the excitation signal is chosen by attempting to match the reconstructed speech waveform as closely as possible to the original speech waveform. A general model for AbS coders is shown in Fig. 1.8. AbS coders were first introduced in 1982 by Atal and Remde with what was to become known as the Multi-Pulse Excited (MPE) coder. Later the Regular-Pulse Excited (RPE), and the Code-Excited Linear Predictive (CELP) coders were introduced. Many variations of CELP coders have been standardized, including [5, 7] G.723.1 operating at 6.3/5.3 kbps, G.729 operating at 8 kbps, G.728 a low delay coder operating at 16 kbps, and all the digital mobile telephony encoding standards including [8, 9, 10] GSM, IS-54, IS-95, and IS-136. The waveform interpolation coder that will be discussed in the subsequent chapters is also a hybrid coder.

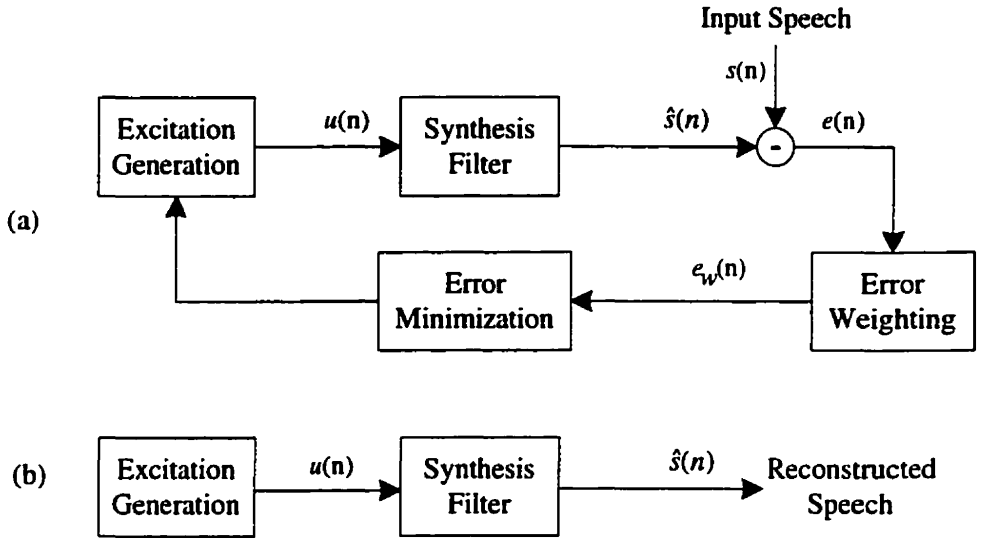


Fig. 1.8 Analysis-by-Synthesis (AbS) coder structure. (a) encoder and (b) decoder.

1.4 Objective and Scope of Our Research

Waveform interpolation (WI), which uses the slowly-evolving waveform (SEW) and rapidly-evolving waveform (REW) decomposition, is widely recognized as a promising low bit-rate encoding technique that overcomes some of the inherent problems of CELP coders to achieve impressive speech quality at rates in the region of 4 kbps. The success of the WI coding scheme is in large part due to its inherent capability of producing an accurate level of periodicity for voiced speech, even at extremely low bit-rates. This contrasts with most CELP-based coders which fail to maintain an appropriate periodicity when operating at about the same rates. Existing WI coders have several shortcomings; some are:

- Significant improvement in reconstructed speech quality is observed when unquantized SEWs are transmitted instead of vector quantized SEWs. This finding indicates that the SEW vector quantization technique in existing WI coder eliminates some perceptually important information.

- Conventional waveform interpolation coders have a modelling error which remains present even when the quantization error is set to zero.

With the above thoughts, this research programme is aimed at improving the performance of existing WI speech encoders by introducing a novel technique in the quantization

of voiced segment (SEW) of speech signal and thereby to bring it closer to toll-quality at 4.25 kbps.

The SEW amplitude spectrum, in the existing WI coder, is split into three non-overlapping subbands: 0–1000 Hz, 1000–2000 Hz and 2000–4000 Hz. The subbands are quantized separately where the baseband is quantized using 8 bits and the remaining two subbands use 3 bits each. Since the dimension of the SEW amplitude spectrum is proportional to pitch period, each band has variable dimension and is quantized using Dimension Conversion Vector Quantization (DCVQ) method. In this thesis we have improved this variable dimension vector quantization issue using the following three approaches:

- The SEW is quantized using a split/shape-gain VQ approach. Instead of using 8 bits for baseband quantization, 6 bits are used. The extra 2 bits are spent to transmit the shape of gain contour of the three bands in each update of SEW.
- A perceptual distortion measure has been introduced to take into account only the audible part of the quantization noise.
- Perceptually-based vector quantization method, which employs the proposed perceptual distortion measure in populating the codebooks, is utilized. The same distortion measure is used to select the best codewords from the codebooks in the process of coding.

Reference [11] presents a novel approach termed Perceptually Irrelevant Phase Elimination (PIPE), to find perceptually irrelevant phase information of acoustic signals. This method is based on the observation that phase change which modifies the relative phase relationship within a critical band is perceptually important, and the phase change, which modifies the phase relationship between critical bands while maintaining phase relationship within a critical band is perceptually irrelevant. It is claimed that this method is applicable for harmonic signal. Slowly Evolving Waveforms (SEWs) contribute to harmonic structure. We evaluate the use of a perceptually relevant SEW phase in the WI coder.

We have implemented and tested a fully quantized simulation of a WI coder incorporating these new features. Subjective tests indicate that the quality of the 4.25 kbps "Improved WI" coder surpasses that of the existing 4.25 kbps WI coder.

1.5 Organization of the Thesis

With the ultimate aim of improving the quality of the WI coder, the present thesis is structured as follows:

In Chapter 2, we will overview the basic theory of linear prediction analysis, specifically short-term linear predictive coding analysis. Conventional methods to obtain the LP coefficients are summarized. Common approaches to represent the spectral parameters are also explained. Different speech-quality-evaluation techniques as applicable to any speech coding algorithm are also described at the concluding section of this chapter.

In Chapter 3, we review the concept and the overall structure of the WI algorithm, with an emphasis on the SEW-REW magnitude quantization layer. We will confine ourselves to Kleijn's frequency-domain approach [12] as implemented by Choy [13].

The focus of Chapter 4 is mainly on quantization theory. The theory of scalar quantization is studied and is then extended to vector quantization. After the principles are reviewed, different product coder structures are presented together with their various degrees of storage and encoding complexity. It describes the implementation of the shape-gain SEW magnitude quantization technique within the WI framework. The second part of this chapter is concerned with the perceptual characteristic of human hearing, confining the focus only to the phase information. The concept of the "perceptually irrelevant phase elimination" [11] method as applicable to any parametric coder including WI model will be investigated.

The simulation results and the comparison with the existing waveform interpolation coder for each of the proposed methods will be presented in Chapter 5.

Finally, the last chapter concludes the thesis with a summary of our work and suggestions for future investigation.

Chapter 2

Linear Prediction of Speech

Like our waveform interpolation coder, most low bit-rate high quality speech coders are based on Linear Predictive Coding (LPC) analysis. The purpose of this chapter is to give an overview of LPC analysis that models the speech signal as a linear combination of its past values and present and past values of a hypothetical input to a system whose output is the given signal. For speech, the prediction is done most conveniently in two separate stages: *short-term* prediction and *long-term* prediction. The short-term prediction captures the near-term correlations while the long-term prediction captures the periodicity of the signal . Specifically, we focus on the short-term LPC. Next, we introduce a popular representation of the LPC coefficients—line spectral frequencies. Finally, subjective and objective distortion measures used to measure the performance of speech coding algorithms are examined.

2.1 Linear Predictive Speech Coding

Linear Predictive Coding (LPC) is the most popular technique for low bit-rate speech coding and has become a very important tool in speech analysis. The popularity of LPC derives from its compact yet precise representation of the speech spectral magnitude as well as its relative simplicity of computation. LPC analysis decomposes the speech into two highly independent components, the vocal tract parameters (LPC coefficients) and the glottal excitation (LP excitation). Here is where the WI scheme comes into play. This scheme is used in efficient representation of the excitation signal and thereby it enhances the coding efficiency.

LPC is not without drawbacks, however. To minimize analysis complexity the LPC

signal is usually assumed to come from an all-pole source; i.e., the assumption is that its spectrum has no zeros. Since the actual speech spectrum has zeros due to the glottal source as well as zeros from the vocal tract response in nasals and unvoiced sounds, such a model is a simplification. The all-pole assumption does not cause major difficulties in speech coders.

2.2 LPC Model

The Linear Predictive Coding (LPC) algorithm of speech compression uses a source-filter model of speech production as has been described in Section 1.3.2 and in Fig. 1.7; this model assumes the vocal tract to be a linear mechanical system which is slowly time varying. The system is modelled as an all-pole (also known as an autoregressive, or AR, model) filter (referred to as the LP synthesis filter) whose input is called an *excitation signal* or *residual signal*, as shown in Fig. 2.1

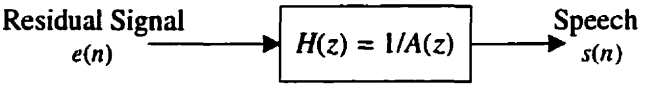


Fig. 2.1 The LP synthesis filter.

Thus, the computation of each speech sample is a linear combination of the previous speech samples and the current excitation to the system. For the current sample, it is necessary to determine if it is voiced or unvoiced. Also, if the sample is voiced, it is necessary to determine the pitch period. The mathematical model for LP coding can be derived with the help of Fig. 1.7 and using the following *relationship* between the physical and the mathematical models:

Vocal Tract	$\Longleftrightarrow$	$H(z)$, LPC filter
Air	$\Longleftrightarrow$	$e(n)$, residual signal or excitation signal or prediction error signal
Vocal Cord Vibration	$\Longleftrightarrow$	V, voiced
Vocal Cord Vibration Period	$\Longleftrightarrow$	T, pitch period
Fricatives and Plosives	$\Longleftrightarrow$	UV, unvoiced
Air volume	$\Longleftrightarrow$	G, gain

The LPC filter is given by:

$$H(z) = \frac{1}{1 - \sum_{k=1}^{p} a_k z^{-k}} = \frac{1}{A(z)} \tag{2.1}$$

which is equivalent to saying that the input-output relationship of the filter is given by the linear difference equation:

$$e(n) = s(n) - \sum_{k=1}^{p} a_k s(n-k), \tag{2.2}$$

where $s(n)$ is the speech signal, the coefficients $a_1, a_2, \ldots, a_p$ are known as LPC coefficients and p is the order of the LP filter. The choice of the order p is a compromise among spectral accuracy, computation time/memory, and transmission bandwidth. In general, it should be selected such that there are at least a pair of poles per each formant. One standard for 8 kHz sampled speech is $p = 10$.

In vector notation, the LPC model can be represented as:

$$\mathbf{L} = [a_1, a_2, \ldots, a_p, T, V/UV, G]. \tag{2.3}$$

The vector $\mathbf{L}$ is assumed to be stationary over a short period of time. In order to meet this slow time varying assumption, we break the speech signal up into frames by multiplying the input signal by a window. Let, the analysis window be $w(n)$ of finite length N_w, the windowed speech segment is given by:

$$s_w(n) = w(n)s(n). \tag{2.4}$$

A Hamming window is commonly used in speech analysis as its tapered edges allow shifting of the analysis frame along the input signal without having large effects on the speech parameters due to the pitch period boundaries or other sudden changes in the speech

signal. The Hamming window is given by:

$$w(n) = \begin{cases} 0.54 - 0.46\cos\left(\frac{2\pi n}{N_w - 1}\right), & 0 \leq n \leq N_w - 1 \\ 0, & \text{otherwise.} \end{cases} \tag{2.5}$$

At a sampling rate of 8 kHz, N_w is selected to be 160 samples i.e., 20 msec, over which the input speech signal is assumed to be stationary. So vector $\mathbf{L}$ is equivalent to,

$$\mathbf{s}_w = [s(0), s(1), \ldots, s(N_w - 1)]. \tag{2.6}$$

Thus the 160 values of $s(n)$ is compactly represented by 13 elements of $\mathbf{L}$.

To get better numerical precision of the LP coefficients the speech signal is often *pre-emphasized* by applying a single-zero filter that increases the relative energy of the high frequency spectrum. That is, the input to LPC analyzer would be the windowed and pre-emphasized version of original input:

$$s_{w\&p}(n) = s_w(n) - \alpha s_w(n-1), \tag{2.7}$$

where α determines the cut-off frequency of the single-zero filter and it is called *pre-emphasis factor*. The typical value of α is around 0.1 [2].

2.3 Estimation of LP Parameters

The linear prediction model is nice in concept for speech synthesis; but its true value is that it is also easy to estimate the parameters from real speech data; i.e., real speech can be approximated by speech generated by the LP model. In order to do so, each of the parameters need to be estimated. This typically consists out of two parts:

- estimation of LP filter parameters
- estimation of LP source parameters (especially pitch)

There are two approaches used for estimating short-term LP filter coefficients $\{a_i\}$:

- autocorrelation method, and

- covariance method.

In our WI implementation the former method is used as the latter method does not guarantee the stability of the all-pole LP synthesis filter. The autocorrelation method is described below:

In the autocorrelation method the LPC coefficients $\{a_i\}_{i=1}^{p}$ are chosen to minimize the energy of the prediction error:

$$E = \sum_{n=-\infty}^{\infty} e^2(n) = \sum_{n=-\infty}^{\infty} \left[s_{w\&p}(n) - \sum_{k=1}^{p} a_k s_{w\&p}(n-k) \right]^2 \tag{2.8}$$

The values of $\{a_k\}$ that minimize E are found by setting $\frac{\partial E}{\partial a_k} = 0$ for $k = 1, 2, 3, \ldots, p$. This yields p linear equations in p unknown coefficients $\{a_k\}_{k=1}^{p}$, also known as the Yule-Walker equations:

$$\sum_{k=1}^{p} a_k \sum_{n=-\infty}^{\infty} s_{w\&p}(n-i) s_{w\&p}(n-k) = \sum_{n=-\infty}^{\infty} s_{w\&p}(n-i) s_{w\&p}(n), \qquad \text{for } i = 1, 2, \ldots, p. \tag{2.9}$$

Defining the autocorrelation function of the windowed and pre-emphasized signal $s_{w\&p}(n)$ as

$$R(i) = \sum_{n=i}^{N_w - 1} s_{w\&p}(n) s_{w\&p}(n-i), \qquad \text{for } 0 \le i \le p. \tag{2.10}$$

Exploiting the fact that the autocorrelation function is an even function i.e., $R(n) = R(-n)$, the set of p linear equations can be represented in matrix form as $\mathbf{Ra} = \mathbf{v}$ which can be rewritten as:

$$\begin{bmatrix} R(0) & R(1) & \cdots & R(p-1) \\ R(1) & R(0) & \cdots & R(p-2) \\ \vdots & \vdots & \ddots & \vdots \\ R(p-1) & R(p-2) & \cdots & R(0) \end{bmatrix} \begin{bmatrix} a_1 \\ a_2 \\ \vdots \\ a_p \end{bmatrix} = \begin{bmatrix} R(1) \\ R(2) \\ \vdots \\ R(p) \end{bmatrix} \tag{2.11}$$

The resulting $p \times p$ autocorrelation matrix is symmetric and Toeplitz. This allows the linear equations to be solved by the Levinson-Durbin algorithm [14]. The Toeplitz structure of $\mathbf{R}$ guarantees that $A(z)$ is minimum phase (zeros inside the unit circle) [15], which in turn ensures the stability of the corresponding LP synthesis filter $H(z) = 1/A(z)$.

2.4 Representation of Spectral Parameters

One of the major issues in LPC is the quantization of the LP parameters [16, 17, 18]. Quantization of the direct form coefficients is generally avoided since quantization error can lead to instability of the LP synthesis filter. On the other hand, quantization of the zeros of $A(z)$ may be done such that the stability of the synthesis filter is ensured. The zeros, however, are difficult to compute. Furthermore, the LPC coefficients, which are typically estimated on a frame level, are needed to be interpolated on subframe level to ensure smoother transition of LPC coefficients over frame-to-frame and for efficient low bit-rate transmission. Direct interpolation of the LPC coefficients can also result in an unstable analysis filter. Therefore, a number of alternate representations of the LPC coefficients have been considered in attempt to find representations which minimize these shortcomings. Some of these representations are: Line Spectral Frequencies (LSF) or equivalently, Line Spectral Pairs (LSP), Reflection Coefficients (RC), ArcSine of Reflection Coefficients (ASRC), Cepstral Coefficients (CC), Log Area Ratios (LAR), AutoCorrelations (AC), and Impulse Response of LP synthesis filter (IR). Despite its computational complexity, LSF is widely used as it provides easy stability checking procedure, spectral manipulations (localized spectral sensitivity) and perceptual quantization for coding. The details of LSF representation are described below:

Line spectral frequencies (LSFs) or line spectral pairs (LSPs) were introduced by Itakura [19]. They represent the phase angles of an ordered set of poles on the unit circle that describes the spectral shape of the LP analysis filter $A(z)$ defined in Eq. (2.1).

Conversion of the LPC coefficients $\{a_k\}$ to the LSF domain relies on $A(z)$. Given $A(z)$, the corresponding LSFs are defined to be the zeros of the polynomials $P(z)$ and $Q(z)$ defined as:

$$\begin{aligned} P(z) &= A(z) + z^{-(p+1)}A(z^{-1}) \\ Q(z) &= A(z) - z^{-(p+1)}A(z^{-1}). \end{aligned} \tag{2.12}$$

It directly follows that:

$$A(z) = \frac{1}{2}\left[P(z) + Q(z)\right]. \tag{2.13}$$

Soong and Juang [20] have shown that if $A(z)$ is minimum phase (corresponding to a stable $H(z)$), all the roots of $P(z)$ and $Q(z)$ lie on the unit circle, alternating between the two

polynomials with increasing frequency, ω . The roots occur in complex-conjugate pairs and hence there are p LSFs lying between 0 and π. The process produces two fixed zeros at $\omega = 0$ and $\omega = \pi$ which can be ignored. It has also been shown [20] that if the p line spectral frequencies ω_i have an ascending ordering property and are unique, then the LP analysis filter $A(z)$ is guaranteed to have minimum phase (stable corresponding synthesis filter):

$$0 < \omega_1 < \omega_2 < \cdots < \omega_p < \pi \quad \text{[radians/sec]}. \tag{2.14}$$

Several approaches for solving for the roots of $P(z)$ and $Q(z)$ have been presented in [20, 21, 22]. Another method by Kabal and Ramachandran [23] makes use of Chebyshev polynomials

$$T_m(x) = \cos(m\omega), \qquad x = \cos\omega. \tag{2.15}$$

The function $x = \cos\omega$ maps the upper semicircle in the z-plane to the real interval $[-1, 1]$. The polynomials $G'(\omega)$ and $H'(\omega)$ can be expanded using the Chebyshev polynomials as follows,

$$G'(x) = 2\sum_{i=0}^{l} g_i T_{l-i}(x), \qquad H'(x) = 2\sum_{i=0}^{m} h_i T_{m-i}(x). \tag{2.16}$$

The roots of these Chebyshev expansions will give the LSFs after the inverse transformation $\omega = \arccos(x)$. The roots are determined iteratively by searching for sign changes of the Chebyshev expansions along the interval $[-1, +1]$.

2.5 Bandwidth Expansion and Lag window

The linear prediction coefficients $\{a_k\}$ parameterize the speech power spectrum. For high pitched voiced signals, since the harmonics are widely spaced, there are too few samples of the envelope spectrum to provide a reliable estimate. For this reason, the formant bandwidths are often underestimated by a large amount resulting in LP synthesis filters with artificially sharp spectral peaks. Either of the following two approaches can be followed to overcome this problem:

- Each LPC coefficient a_k is replaced by $\gamma^k a_k$. As a result, all the poles of $H(z)$ move inward by a factor γ and this causes a *bandwidth expansion* of the formant peaks in the frequency response. The typical values for γ are between 0.988 and 0.996 which correspond to 10 to 30 Hz bandwidth expansion.

- Another approach to expand the estimated formant bandwidth is to multiply the autocorrelation coefficients by a lag window prior to the computation of LP parameters. The lag window is often chosen to have a Gaussian shape. It is equivalent to convolving the power spectrum with a Gaussian shape window and this widens the peaks of the spectrum.

2.6 Speech Quality and Evaluation

A speech coding algorithm is evaluated based on the following attributes: (*i*) the bit rate of the compressed signal, (*ii*) the quality of the reconstructed ("coded") speech, (*iii*) the complexity of the algorithm, (*iv*) the delay introduced, and (*v*) the robustness of the algorithm to channel errors and acoustic interference. Different applications require the coder to be optimized for different features or some balance between these features. In general high-quality speech coding at low-rates is achieved using high complexity algorithms. On the other hand, a real-time implementation imposes constraints on both the complexity and the delay of the coder. In message transmission systems, for instance, the delay of the coder may not be an issue, and central storage systems may not require a low-complexity implementation of the coder. While in a large number of applications the primary goal is to ensure the perceived similarity between the original and the reconstructed signal, in some cases (i.e., in the systems in which security is the main concern) it is sufficient that the reconstructed speech sounds intelligible and natural. Moreover, in some applications coders must perform reasonably well with speech corrupted by background noise, non-speech signals (such as DTMF tones, voiceband data, modem signals, etc.), and a variety of languages and accents.

In digital communications speech quality is classified into four general categories, namely:

- *commentary or broadcast* quality refers to wide-bandwidth (typically 50–7000 Hz, but 20–20,000 Hz for compact disk) high-quality speech that can generally be achieved at rates, at least 32–64 kbps.

- *network or toll or wireline* quality describes speech as heard over the switched telephone network (approximately the 200–3200 Hz range, with a signal-to-noise ratio of more than 30 dB and with less than 2–3% harmonic distortion).

- *communications* quality implies somewhat degraded speech quality which is nevertheless natural and highly intelligible. Communications speech can be achieved at rates above 4 kbps.

- *synthetic* speech is usually intelligible but can be unnatural and associated with a loss of speaker recognizability.

The current goal in speech coding is to achieve toll quality at 4.0 kbps. Currently, speech coders operating well below 4.0 kbps tend to produce speech of synthetic quality.

Gauging the speech quality is an important but also very difficult task. There are two typical ways to measure the speech quality:

- Subjective speech quality measures.

- Objective speech quality measures.

2.6.1 Subjective Speech Quality Measures

A subjective evaluation procedure is usually done using listening tests with response set of syllables, words, sentences, or with other questions. The test material is usually focused on consonants, because they are more difficult to synthesize than vowels. Especially nasalized consonants (/m/, /n/, /ng/) are usually considered the most problematic. The other difficult consonants and their combinations are for example /d/, /g/, /k/, /lb/, /dr/, /gl/, /gr/, /pr/, /spl/, /rp/, /rt/, /rch/, and /rm/. In these tests speech quality is usually measured by *intelligibility* typically defined as the percentage of words or phonemes correctly heard, and *naturalness*. There are three types of commonly used subjective quality measures.

- *Diagnostic Rhyme Test* (*DRT*): The DRT is an intelligibility measure where the subject's task is to recognize one of two possible words in a set of rhyming pairs (e.g., meat - heat) [24].

- *Diagnostic Acceptability Measure* (*DAM*): The DAM scores evaluate the quality of a communication system based on the acceptability of speech as perceived by a trained normative listener.

- *Mean Opinion Score (MOS)*: The MOS is a measure which is widely used to quantify coded speech quality. The MOS usually involves 12 to 24 listeners [25] (formal CCITT and TIA tests typically involve 32–64 listeners) who are instructed to rate phonetically balanced records according to a 5-level quality scale. The rating scale and its description is presented in Table 2.1 and Fig. 2.2.

Table 2.1 Description of the Mean Opinion Score (MOS) [26].

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

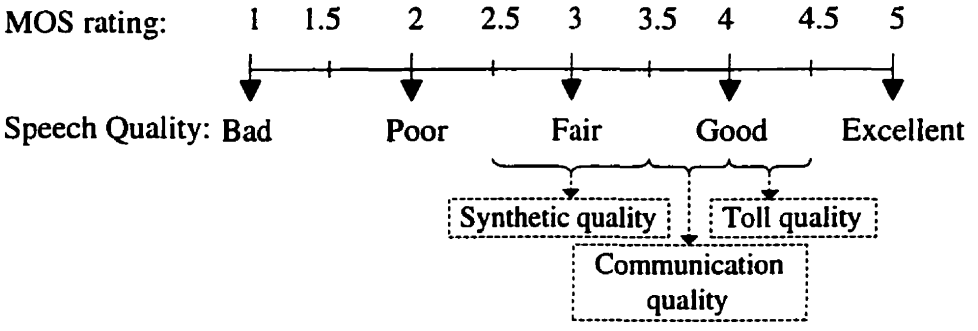


Fig. 2.2 Schematic representation of the relation of MOS rating and speech quality.

We note here that MOS ratings may differ significantly from test to test and hence they are not absolute measures for the comparison of different coders.

2.6.2 Objective Speech Quality Measures

The human auditory system is the ultimate evaluator of the quality and performance of a speech coder in preserving intelligibility and naturalness. While extensive subjective listening tests provide the most accurate assessment of speech coders, they can be time consuming and inconsistent. Objective measurements can give an immediate and reliable estimate of the perceptual quality of a coding algorithm [26].

Objective Distortion Measures in the Time Domain

The followings are the major types of time domain objective distortion measures:

- *Signal-to-Noise Ratio (SNR)*: The is one of the most common objective measures for evaluating the performance of a compression algorithm. SNR is defined as the ratio of the average speech energy to the average energy in the error signal, and is usually expressed in decibels as $10 \log_{10}$ SNR. The SNR is a long-term measure for the accuracy of speech reconstruction and as such it tends to "hide" temporal reconstruction noise particularly for low level signals.

- *Segmental SNR (SEGSNR)*: Temporal variations of the performance can be better detected and evaluated using a short-time (frame-by-frame basis) signal-to-noise ratio. The frame-based measure is called the segmental SNR (SEGSNR). For each frame (typically 15–25 msec), an SNR measure is computed and the final measure is obtained by averaging these measurements over all segments of the waveform. Since the averaging operation occurs after the logarithm, the SEGSNR penalizes coders whose performances vary.

It is important to note that SNR-based measures (e.g., SNR or SEGSNR) are only appropriate for coding systems that seek to reproduce the original input waveform (e.g., waveform coders). These measures are usually simple, but cannot be used for vocoders due to their time-domain evaluation, which requires temporal synchronization (lost in vocoders) [2]. A frequency-dependent SNR can be computed by filtering the signals through a filter bank and computing the SNR for each frequency band [4].

Objective Distortion Measures in the Spectral Domain

In the frequency domain, the LPC spectrum of the original signal and the LPC spectrum of the quantized or interpolated signal are compared. A spectral distortion measure should be capable of tracking the distortion or difference between the two spectra that affects the perception of the sound. In the following situations, the disparities between the original and coded spectral envelopes can perceptually lead to sounds that are phonetically different:

- The formants of the original and coded spectral envelopes occur at significantly different frequencies.

- The bandwidth of the formants of these spectral envelopes differ significantly.

A brief description of different types of spectral distortion measures is presented below.

- *Itakura Measure*: The Itakura measure [19] generally corresponds better to the perceptual quality of speech. Also known as a likelihood ratio distance measure, this is the most widely used measure for LP vocoders. This measure is based on the similarity between all-pole models of the reference and coded speech waveforms. The distance measure is computed between sets of LP parameters estimated over synchronous frames (typically every 15–30 msec) in the original and processed speech. The Itakura measure is heavily influenced by spectral error due to mismatch in formant locations, whereas errors in matching spectral valleys do not contribute heavily to the distance. This is desirable, since the auditory system is more sensitive to errors in formant location and bandwidth than to the spectral valleys between peaks.

- *Log Spectral Distortion Measure*: The most frequently used spectral distortion measure is termed the Root Mean Square (RMS) log spectral distortion, or simply the *spectral distortion*. Spectral distortion for a given frame is defined as the root mean square difference between the original LPC log power spectrum and the quantized or interpolated LPC log power spectrum. Usually the average of spectral distortion over a large number of frames is calculated, and that is used as the measure of performance of quantization or interpolation.

- *Weighted Euclidean Distance Measure*: This measure is performed in the LSF domain or in the magnitude spectrum domain. In our research we use this in the codebook search in a perceptually efficient manner. A detailed description of this distortion measure is presented in Section 4.1.4.

Other objective measures often mentioned in the literature include the log-area ratio measure, cepstral distance and articulation index.

2.6.3 Objective Measures Predicting the Subjective Quality of Speech

Objective measures are often sensitive to both gain variations and delays. More importantly, they typically do not account for the perceptual properties of the ear [27]. On the other hand, formal subjective evaluations, such as the ones described above, can be lengthy

and very costly. Recent efforts in speech quality assessment are focused upon developing automatic test evaluation procedures and objective measures that are capable of predicting the subjective quality of speech [28, 29]. They use two signals as their input, namely an original signal (reference pattern) and the corresponding output signal after its transition through the speech coder under test as shown in Fig. 2.3. The signal processing within

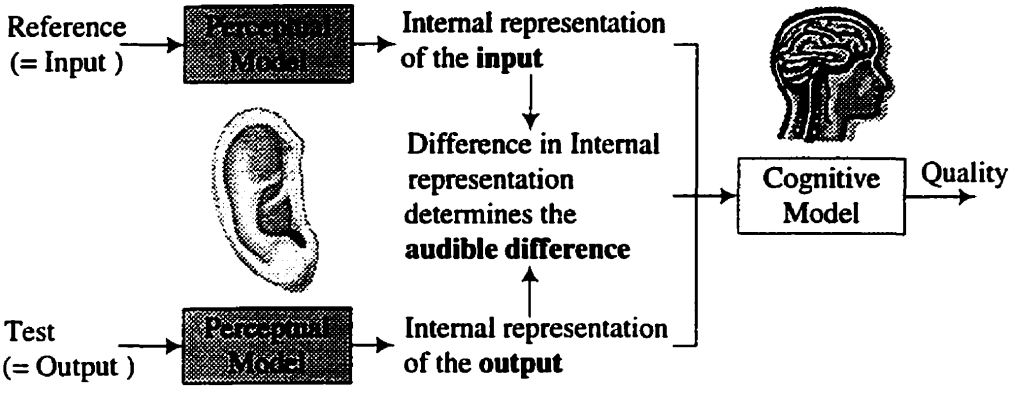


Fig. 2.3 The basic model approach of objective measure of speech quality based on perceptual quality measurement.

the perceptual measures can be structured into three major steps: pre-processing [30, 31], psycho-acoustic modelling, and cognitive modelling. The cognitive modelling is the one that mostly differentiates the objective methods. The cognitive model assesses the subjective judgement of the speech quality. The assessment is accomplished by determining a perceptual distance between the measured signal and the reference and then by creating a figure of merit that describes the speech quality. The figure of merit is generally a non-linear function of the subjectively determined MOS value. In order to obtain an objective estimator for the MOS value, it is necessary to map the objective result to the MOS scale.

The most known algorithms for objective speech quality evaluation based on a psycho-acoustic sound perception model are: BSD (Bark Spectral Distance) [26], PSQM (perceptual Speech Quality Measure and ITU-T P.861 standard 1997) [32].

2.7 Summary

In this chapter, we have discussed how the speech signal can be divided into two highly independent components—LPC coefficients and LP excitation, using LP analysis. The next

chapter will discuss how the LP excitation can be coded in a complex but efficient manner using the Waveform Interpolation (WI) model.

Chapter 3

Waveform Interpolation Coder

3.1 Background and Overview

The quality of waveform coders, such as Code-Excited Linear Predictive (CELP) coders and their derivative architectures, degrades rapidly at rates below 4.8 kbps. This is primarily due to the fact that the CELP algorithm relies on a waveform-matching procedure, which essentially maximizes the (spectrally weighted) signal-to-noise ratio (SNR) on a frame-by-frame basis. However, SNR is not an ideal measure of the perceptual quality of the reconstructed speech signal. It has been shown that by increasing the pitch periodicity the perceptual quality of the CELP encoded speech signal can be improved at the cost of SNR [33]. At low bit-rates the conventional CELP structure cannot preserve the correct degree of periodicity of the speech signal, ultimately resulting in a noisy reconstructed signal. Thus, for effective high perceptual-quality coding of speech at 4 kbps, it is necessary to develop a coding algorithm with inherent periodicity. Further, to achieve low bit-rates, while maintaining quality, the algorithms must exploit the evolutionary nature of speech. The Waveform Interpolation (WI) paradigm [34] offers both of these properties by representing speech or, more often, the LP residual as an evolving set of pitch-cycle waveforms [35]. For voiced segments, the pitch-cycle waveforms describe the essential characteristics (pitch cycle of nearly periodic segment) of the speech signal and hence are known as *prototype* or *characteristic waveforms* (*CW*). At low bit-rates the shape of the pitch-cycle waveforms evolves slowly as a function of time. This suggests that pitch-cycle waveforms can be extracted at infrequent time intervals. Interpolation can then be exploited to obtain an approximation of the intermediate pitch-cycle waveforms. This reasoning was the original motivation for

speech-coding using WI. It was first introduced by Kleijn [12] and the first version was called Prototype Waveform Interpolation (PWI). PWI leads to efficient coding of voiced speech, but it switches to CELP for non-periodic unvoiced signals. In 1994, the PWI paradigm was extended to provide an effective basis for the coding of voiced and unvoiced speech and background noise. In this extended WI coding scheme, each phase-aligned prototype waveform derived from the speech/residual is decomposed into nearly-independent *Slowly Evolving Waveforms* (*SEWs*) characterizing the voiced part, and a remainder, the *Rapidly Evolving Waveforms* (*REWs*) representing noise-like unvoiced speech. Because of its low bandwidth, a low bit rate suffices for coding the SEW (additional processing lowers the bit rate further), while the REW requires only a rough statistical description (e.g., its phase spectrum can be randomized).

The method of describing the pitch of the speech is reminiscent of first-generation vocoding algorithms but, while WI utilizes many familiar concepts such as LP coding and subsequent LSF quantization, the majority of the concepts are new to speech coding. Further, the technique attains high quality at low bit-rates by utilizing smooth interpolation of almost all of its parameters paying special attention to events such as pitch doubling. The technique is a truly hybrid speech coding algorithm, performing analysis in both the time and the frequency domains. In particular, the use of Fourier descriptions of the prototypes allows effective phase alignment and interpolation between characteristic waveforms of varying pitch.

Fig. 3.1 provides an overview of the WI scheme as implemented by Choy [13]. It can be divided into two layers: the *outer layer* and the *inner layer*. The outer layer defines the basic analysis-synthesis system, whereas the inner layer performs the quantization and the dequantization of the coder parameters. The outer layer converts the one-dimensional input signal into a two-dimensional CW at the analyzer (processor **100**) and the inverse process is performed at the synthesizer (processor **200**). The inner layer operates on a two-dimensional signal, which represents the waveform shape along one axis (ϕ axis), and the evolution of the shape along the time axis (t axis) as shown in Fig. 3.2.

Processor **300** decomposes CW into SEW-REW and quantizes these parameters in a perceptually efficient manner while processor **400** performs the inverse process. Processor **100** and processor **300** form the WI encoder and the other two processors form the WI decoder. Section 3.2 will describe the WI encoder and Section 3.3 will dissect the WI

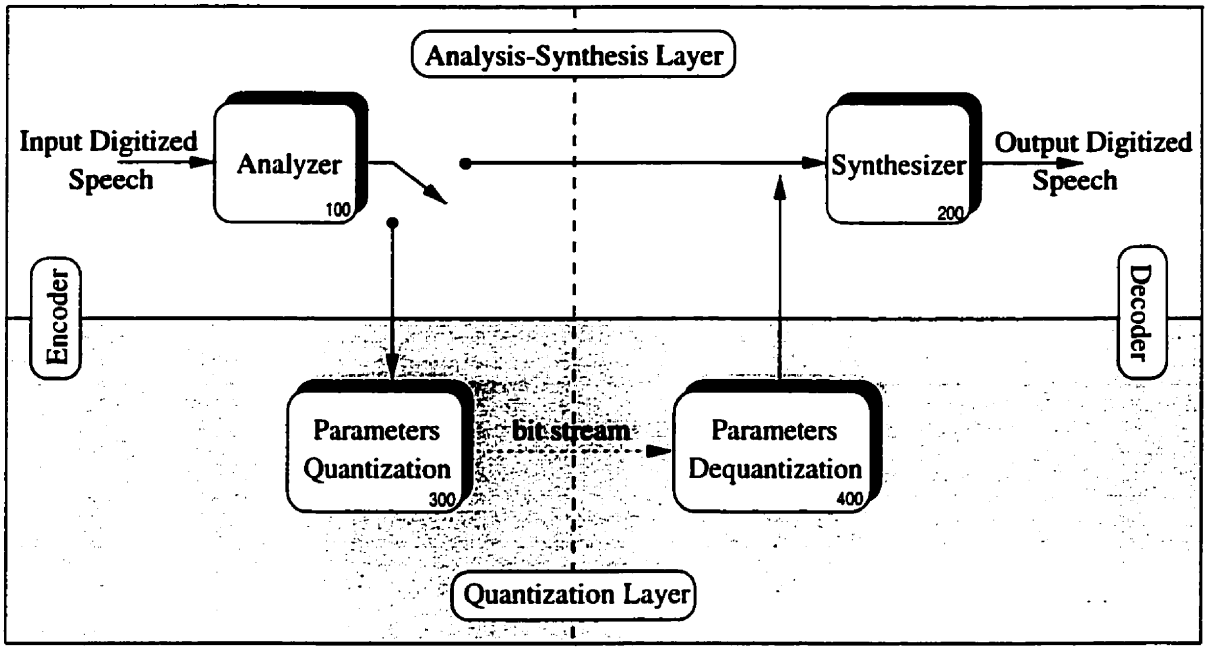


Fig. 3.1 Overview of the WI method. The switch facilitates the coder to bypass the quantization layer and thereby allows us to evaluate the performance of the analysis-synthesis layer[1]. (From [13].)

decoder.

3.2 WI Encoder

Like other coders belonging to the *linear predictive analysis-by-synthesis* coder family, in traditional open-loop WI coding, the speech signal is first converted to the residual signal via a linear-predictive (LP) analysis filter. The LP parameters, derived by autocorrelation method in 10th order LP analysis with an update rate of 50 Hz, are quantized as LSF vectors using a split vector quantization algorithm.

The residual speech signal is then transmitted to a two-dimensional characteristic waveform (CW) surface obtained by extraction and subsequent alignment of pitch-length characteristic waveforms. By filtering this surface along the time axis, the surface (and hence the evolution of the characteristic waveforms) is decomposed into two underlying compo-

[1]For the purpose of clarity, each functional block in the WI schematic diagram is referred to as a processor and is identified by a three-digit number. The schematic diagrams for processor **100** and **200** are shown in Fig. 3.3 and Fig. 3.9 respectively while that of processor **300** and **400** are shown in Fig. 3.7.

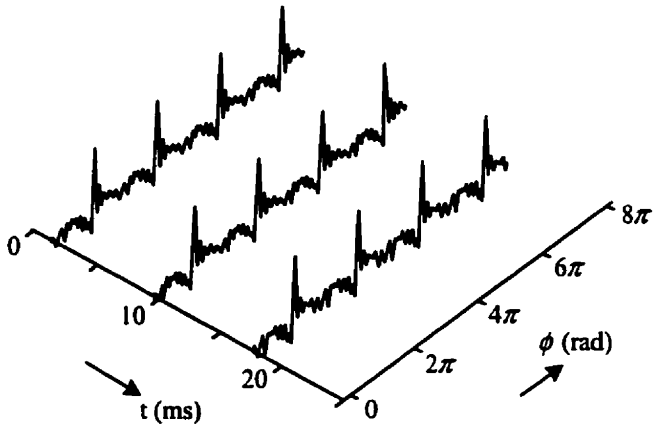


Fig. 3.2 The concept of the ϕ axis and the t axis in the two-dimensional CW representation of the speech signal.

nents, the rapidly evolving waveform (REW) and the slowly evolving waveform (SEW). In fact, the SEW results from lowpass filtering the CW surface and the highpass filtering operation leads to the REW.

The logarithmic power of the speech waveform is encoded with a non-adaptive differential scalar quantizer using a 4-bit codebook and transmitted at a rate of 100 Hz (twice per frame).

The analysis part of the WI encoder architecture is shown in Fig. 3.3

3.2.1 Characteristic Waveform Extraction

After the pitch is determined using the pitch estimation algorithm followed in EVRC (Enhanced Variable Rate Codec) [36] (and quantized to 7 bits) and interpolated in the subframe level the one-dimensional characteristic waveform (CW) is extracted at a rate of 400 Hz (8/frame). The CW extraction is performed by first dividing the current frame into eight segments of equal length. The endpoint of each segment is marked by an *extraction point* as shown in Fig. 3.4(a). At each extraction point a square window, commonly known as *extraction window*, having the length of the interpolated pitch period of that subframe is centered and the residual segment bounded within the window is the extracted CW. The window location is not tightly constrained, but instead the location is determined so that the window boundaries are located in regions of relatively low power. In order to efficiently determine the position of window boundary the position of each extraction point is relaxed

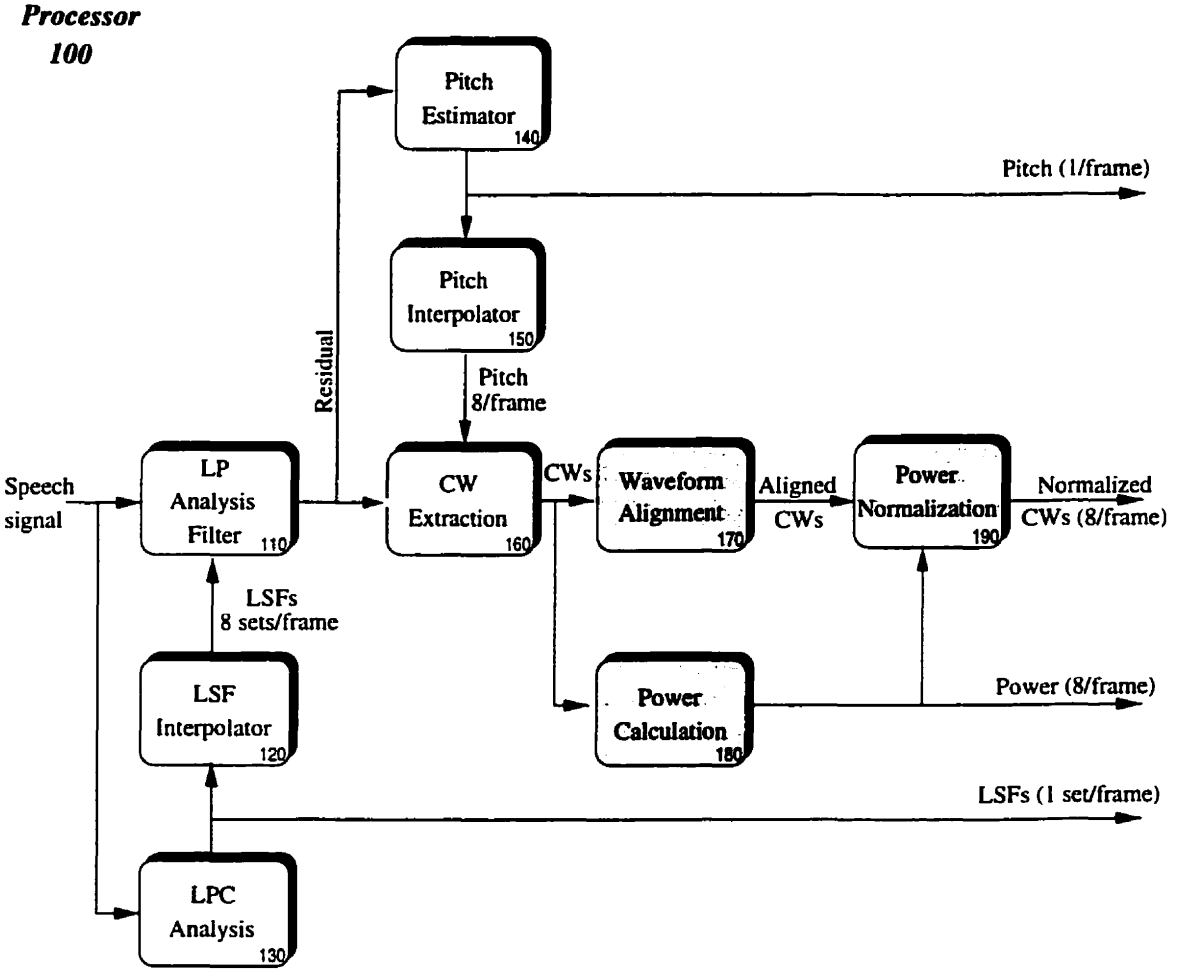
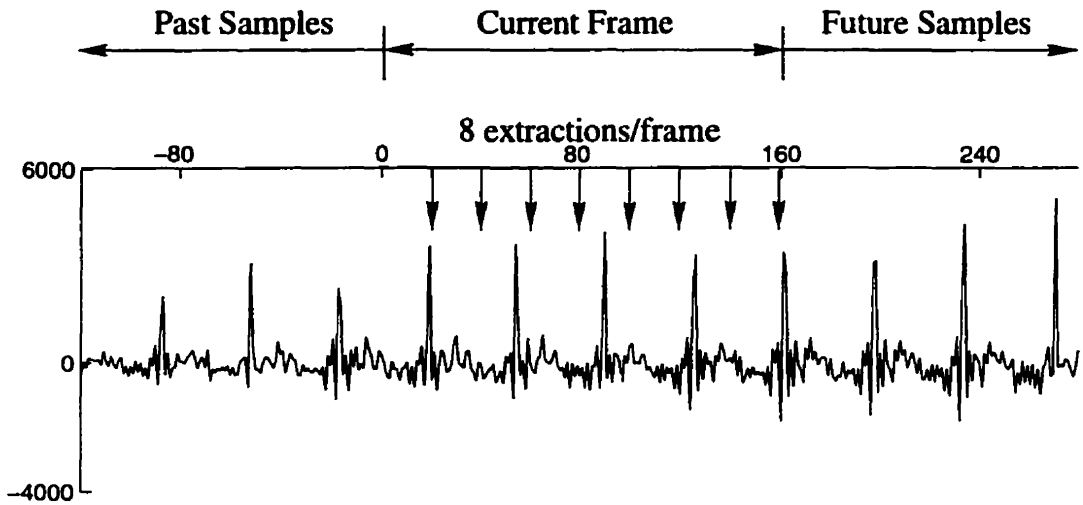
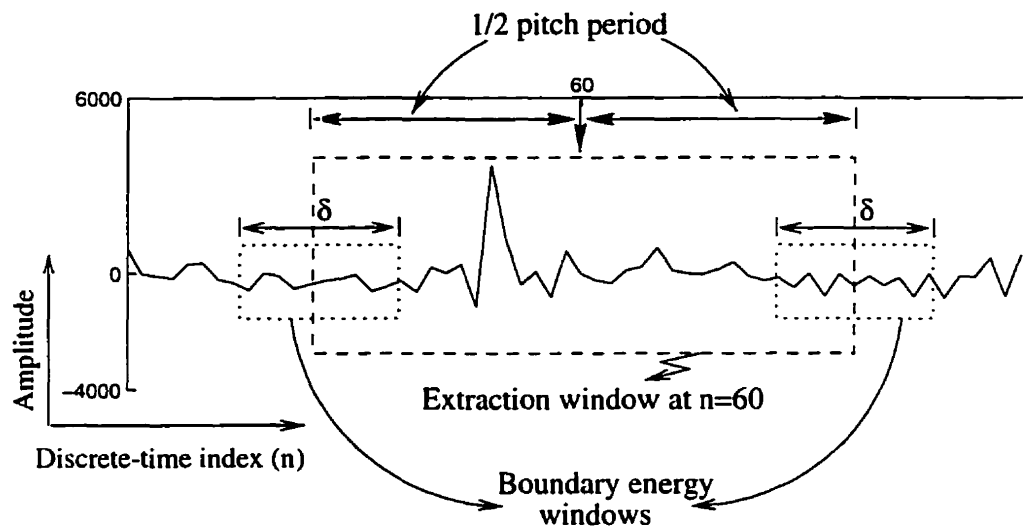


Fig. 3.3 A block diagram of the analysis part of WI Encoder (processor **100**). The update rate of the shaded processors is once per subframe while for non-shaded processors it is once per frame. (From [13].)



(a) A segment of an LP-residual signal, spoken by a male speaker.



(b) Illustration of an extraction window and its boundary energy windows.

Fig. 3.4 An example of CW extraction. (a) shows the original locations of the eight equally spaced extraction points for a frame of residual signal. (b) shows how to extract CW by relaxing the position of extraction points in such a way that the window boundaries are located in regions of relatively low power. (From [13].)

by a value of ε. The maximum value of ε is 16 samples [13]. Now, another window called *boundary energy window* is created which centers at each side of extraction window. The length of the boundary energy window is varied by a value of δ to get the minimum energy position for boundaries of extraction window. The maximum value of δ is 10 samples [13].

The CWs are ultimately used to construct a two dimensional surface $u(n, \phi)$ to display the shape of the discrete time waveform along the ϕ axis and the evolution of the shape along the n axis. The mathematical basis for the formation of the two-dimensional evolving surface is presented below:

The Discrete Time Fourier Series (DTFS) representation of a single, one-dimensional CW is given by:

$$u(m) = \sum_{k=0}^{\lfloor P/2 \rfloor} \left[A_k \cos\left(\frac{2\pi km}{P}\right) + B_k \sin\left(\frac{2\pi km}{P}\right) \right], \qquad 0 \leq m < P \tag{3.1}$$

where A_k and B_k are the DTFS coefficients.

To construct the two-dimensional representation for a sequence of CWs, it is convenient to consider the CW as one cycle of periodic function of ϕ and to normalize the pitch period of this periodic function to 2π. Because the original signal is band limited, the periodic function of ϕ can be obtained from the above equation by introducing a discrete-time index n. With the thought that the DTFS coefficients and pitch periods are now time-varying, the expression for the two-dimensional CW can be written as follows.

$$u(n, \phi) = \sum_{k=1}^{\lfloor P(n)/2 \rfloor} \left[A_k(n) \cos(k\phi) + B_k(n) \sin(k\phi) \right] \qquad 0 \leq \phi(\cdot) < 2\pi \tag{3.2}$$

where

$$\phi = \phi(m) = \frac{2\pi m}{P(n)} \tag{3.3}$$

Note that the lower limit of the summation runs from $k = 1$. This is because B_0 vanishes automatically since $\sin(0) = 0$, and A_0 is ignored as it represents the DC component having no perceptual significance. Once CWs are extracted in processor **160** they need to be time-aligned. The following subsection discusses the CW alignment procedure.

Most WI architectures rely on Kleijn's frequency-domain approach [33, 37] as has been followed in our coder too (originally implemented by Choy [13]), although there are schemes,

such as that proposed by Hiotakakos and Xydeas [38] as well as Hiwasaki and Mano [39], which employ time-domain coding using time-domain representation of CWs.

3.2.2 CW Alignment

Once the characteristic waveform is extracted from the residual signal, the smoothness of the surface $u(n,\phi)$ in the n direction must be maximized. This can be accomplished by alignment, in ϕ, of the extracted CW with the previously extracted CW by introducing a circular time shift to the current one. The circular time shift is indeed equivalent to adding a linear phase to the DTFS coefficients. To get a basic understanding of the alignment criterion, let us begin with the DTFS representation of a pair of successive unaligned CWs:

$$\begin{aligned}\text{previous CW:}\quad u(n_i-1,m) &= \sum_{k=1}^{\lfloor P(n_i-1)\rfloor}\left[A_k(n_{i-1})\cos\left(\frac{2\pi km}{P(n_i-1)}\right)\right.\\ &\qquad\left.+\,B_k(n_{i-1})\sin\left(\frac{2\pi km}{P(n_i-1)}\right)\right]\\ \text{current CW:}\quad u(n_i,m) &= \sum_{k=1}^{\lfloor P(n_i)\rfloor}\left[A_k(n_i)\cos\left(\frac{2\pi km}{P(n_i)}\right)+B_k(n_i)\sin\left(\frac{2\pi km}{P(n_i)}\right)\right]\end{aligned} \tag{3.4}$$

For convenience it is assumed that these two CWs are of equal dimension i.e.,

$$\begin{aligned}P(n_i) &= P(n_i-1) = P\\ \lfloor P(n_i)/2\rfloor &= \lfloor P(n_i-1)/2\rfloor = K.\end{aligned} \tag{3.5}$$

Suppose now a circular time shift of T samples (to the right) is applied to the current CW, $u(n_i,m)$ resulting in $u(n_i,m-T)$ which is a linear phase shifted version of $u(n_i,m)$ shifted by an amount of $2\pi T/P$. The amount of time shifting T, or equivalently the corresponding phase shifting ϕ to align $u(n_i,m-T)$ with $u(n_i-1,m)$ is obtained by maximizing the cross-correlation between the two CWs. The estimated aligned characteristic waveform is then

$$\hat{u}(n_i,\phi) = u(n_i,\phi-\phi_u) \tag{3.6}$$

Here, ϕ_u is the amount of optimal phase shift required for alignment, and it is given by

$$\phi_u = \arg\max_{0\le\phi_e<2\pi} \sum_{k=1}^{K} \{[A_k(n_{i-1})A_k(n_i) + B_k(n_{i-1})B_k(n_i)]\cos(k\phi_e) + [B_k(n_{i-1})A_k(n_i) - B_k(n_i)A_k(n_{i-1})]\sin(k\phi_e)\}. \quad (3.7)$$

A detailed discussion on the above alignment criterion can be found in [40].

After the CWs are extracted and aligned, their powers are normalized. The objective of separating gain from the CW vector is that it reduces the pattern of variations in a vector and thereby improving coding efficiency. Fig. 3.5 shows the CW surface after their power is normalized and length is normalized to a length of 2π. The surface illustrates the characteristic waveform features (one pitch-cycle each) as a function of the phase along the ϕ-axis, and the evolution of the waveforms along the time axis. It is obvious from Fig. 3.5 that for the unvoiced part of the speech signal the CW surface evolves rapidly, and for the voiced segment the evolution of the surface is relatively slower.

3.2.3 CW Decomposition

The characteristic waveform representation is particularly convenient for separation into voiced and unvoiced components. For the voiced component of the original signal (the quasi-periodic component), the CW evolves slowly as a function of time. In other words, the correlation between CWs decays slowly with increasing separation. In contrast, for the unvoiced component of speech the correlation between CWs vanishes when their separation is one pitch period or more, and as a result, the CW evolves rapidly. This property of the CWs suggests that low pass filtering the CW surface (i.e., the DTFS coefficients along the time axis) leads to a slowly evolving waveform (SEW). The rapidly evolving part of the signal (REW) can be found by subtracting the SEW from the CW i.e.,

$$u_{REW}(n,\phi) = u_{CW}(n,\phi) - u_{SEW}(n,\phi). \quad (3.8)$$

Fig. 3.6 illustrates the decomposition of the characteristic waveform shown in Fig. 3.5 into SEW and REW. The linear filter used for the decomposition is a linear phase, non-causal, FIR filter with a cut-off frequency of 25 Hz. The perception of vowels is affected when the envelope of the spectrum is smeared by lowpass filtering with a lowpass filter of 16 Hz or lower [41], suggesting that the cut-off frequency of the filter should be around 20 Hz. So

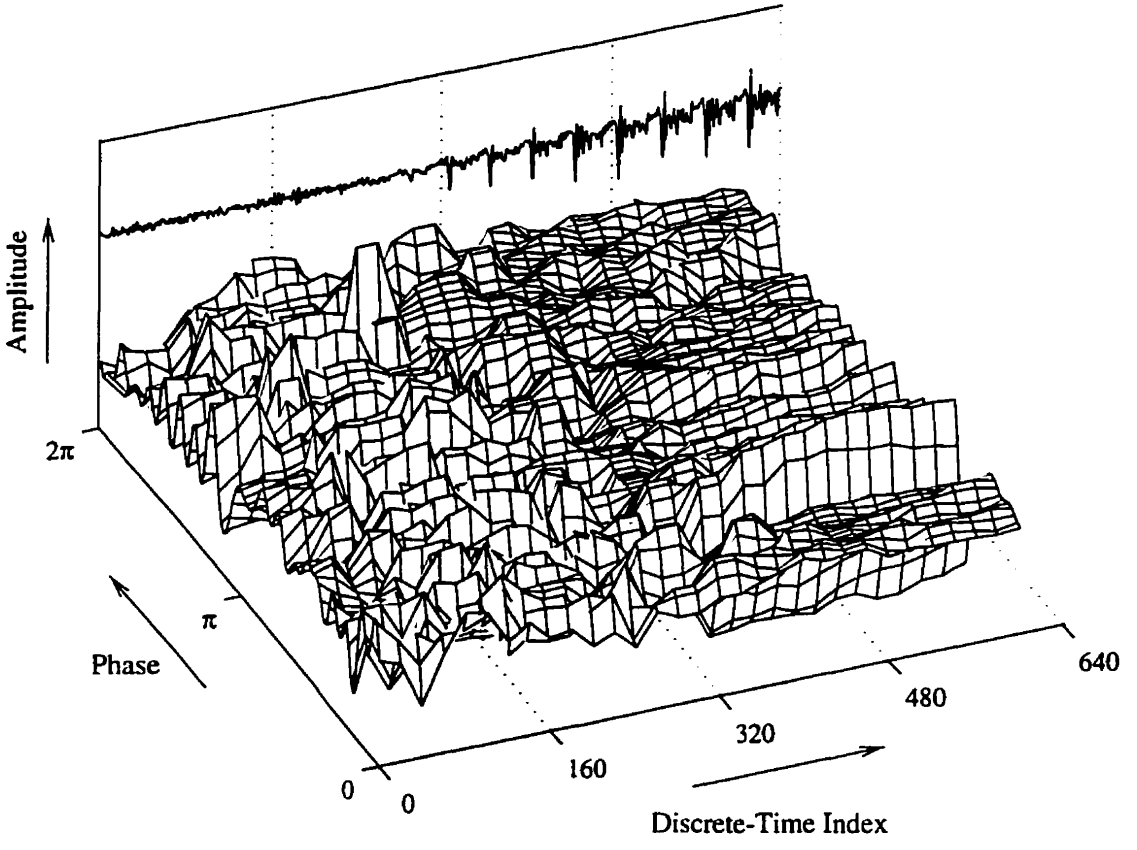


Fig. 3.5 An example of a characteristic waveform surface, $u(n, \phi)$. The surface displays the shape of the discrete time waveform along the phase (ϕ) axis and the evolution of the shape along the discrete-time (n) axis. It is derived by decomposing the residual signal for the word "help" spoken by a male speaker. (From [13].)

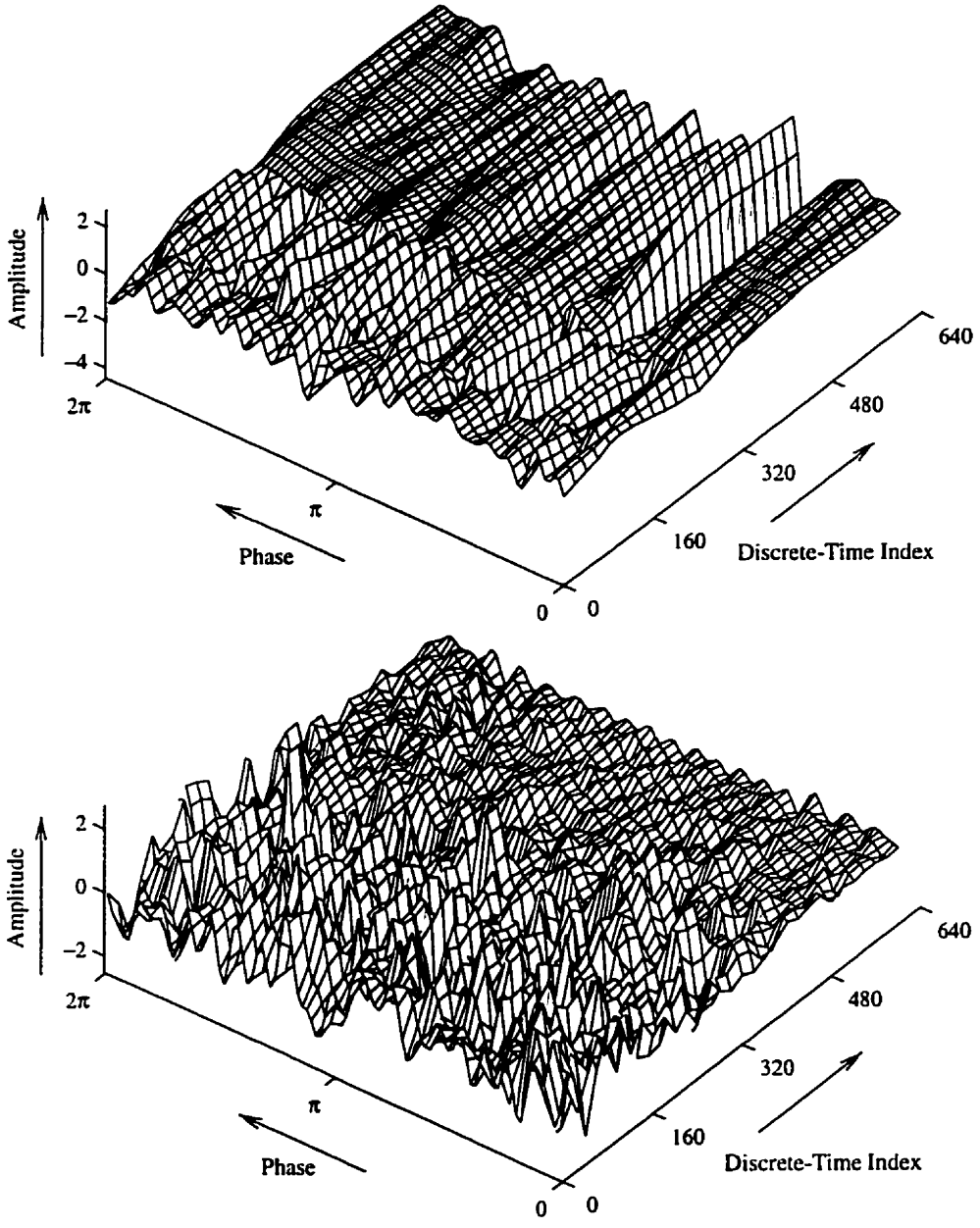


Fig. 3.6 The SEW (top) and the REW surfaces (bottom) for the CW shown in Fig. 3.5. The SEW is obtained by lowpass filtering the CW surface and by subtracting SEW from the CW we get REW. (From [13].)

25 Hz is a safe choice. To avoid the inclusion of extra coder delay, this lowpass filter uses 17 taps corresponding to a delay of $(8 \times (1/8) =)$ 1 frame.

3.2.4 Quantization of WI Parameters

Using the features described above, the WI paradigm leads to highly independent ("orthogonal") parameters: (*i*) pitch, (*ii*) LP parameters (LSFs), (*iii*) signal power (gain), and (*iv*) CW (REW and SEW). The independence means that the design of the quantizers is significantly simplified. Fig. 3.7 shows the block diagram of the WI quantizers and the processors involved in this job. The transmission rate for the pitch is once per frame (50

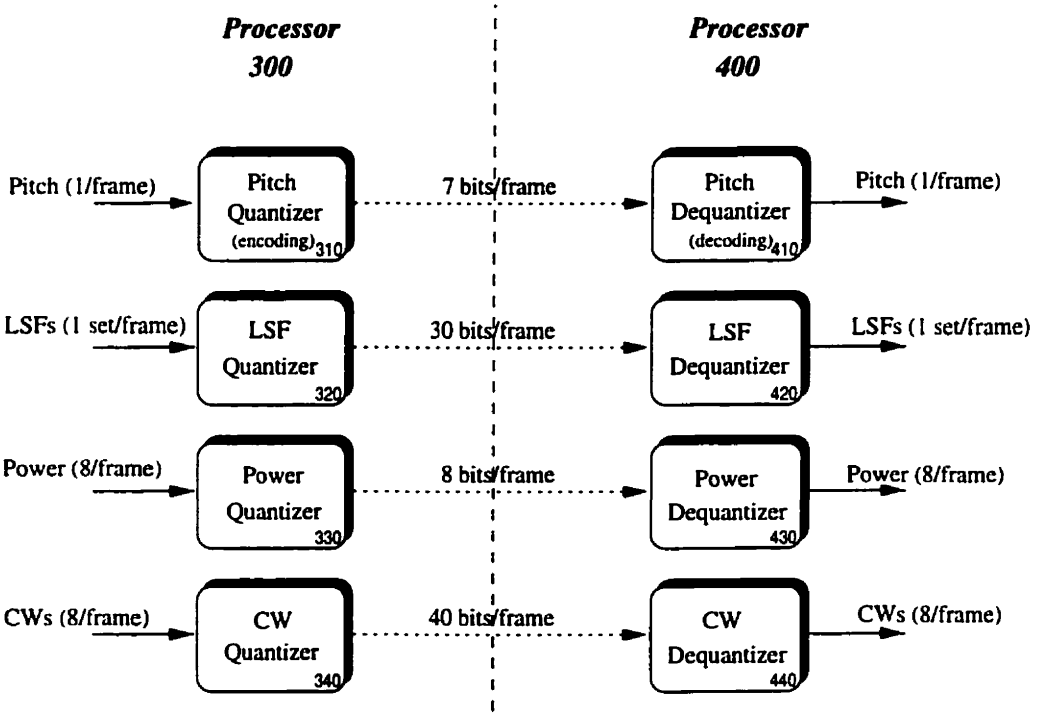


Fig. 3.7 A schematic diagram of the WI quantizers and dequantizers. The internal blocks of processors **330, 430** and processors **340, 440** are shown in Fig. 3.8. (From [13].)

Hz). We use 7 bits to quantize pitch. Since our pitch estimation algorithm generates only integer pitch values within a range of 20 to 120, no quantization error is introduced for the pitch value.

The LP parameters are quantized as LSF vectors using a 30-bit split vector quantization algorithm. In this split VQ algorithm a vector of 10 LSFs is divided into three sub-vectors

of dimension 3, 3 and 4, each of which is quantized using 10 bits. The best codeword is selected based on the minimum weighted Euclidean distance [42] which tries to assign weights to individual LSFs according to their spectral sensitivity.

The quantization of the power and CWs require further processing prior to coding. Fig. 3.8 illustrates the quantization and dequantization steps of power and CW (SEW and REW) along with the preprocessing steps.

SEW Quantization

The perception of purely voiced sounds and purely unvoiced sounds differs greatly. So the separation of the evolving CW into a SEW and a REW allows efficient perceptual based coding of the CW. The human-auditory system is very sensitive to small changes in the spectrum of the quasi-periodic component (SEW) of the speech signal. This property in perception suggests that a precise description of the SEW should be transmitted with a higher number of bits. On the other hand, SEW evolves slowly which means that its inherent information rate is relatively low. This suggests that SEW can be described accurately at a low bit-rate, allowing down-sampling and/or differential quantization. Typically, the SEW is downsampled to a rate of 100 Hz (two SEWs per frame) as shown in Fig. 3.8. The amplitude spectrum of each downsampled SEW is split into three non-overlapping subbands: 0–1000 Hz, 1000–2000 Hz and 2000–4000 Hz. The first subband is quantized using 8 bits and the other two bands are quantized separately using 3 bits each. Since the length of the SEW depends on the interpolated pitch period in that subframe which is not constant, a variable dimension vector quantizer (VDVQ) is employed for each subband. The GLA technique (to be discussed in Section 4.1.3) is applied to design the codebook for each subband. The search process for the best codeword within the first codebook incorporates perceptual weighting effects. The perceptual weighting simulates the spectral masking property in human-auditory system which allows more quantization noise in the peak regions than in the valleys in the formant structure of the speech signal. In the residual signal driven WI coder, the weighting factor, γ_w, is absorbed into the synthesis filter [4].

$$H_w(z) = \frac{1}{1 - \sum_{k=1}^{p} a_k(\gamma_w)^k z^{-k}} = \frac{1}{A(z/\gamma_w)}, \qquad 0 < \gamma_w \leq 1. \tag{3.9}$$

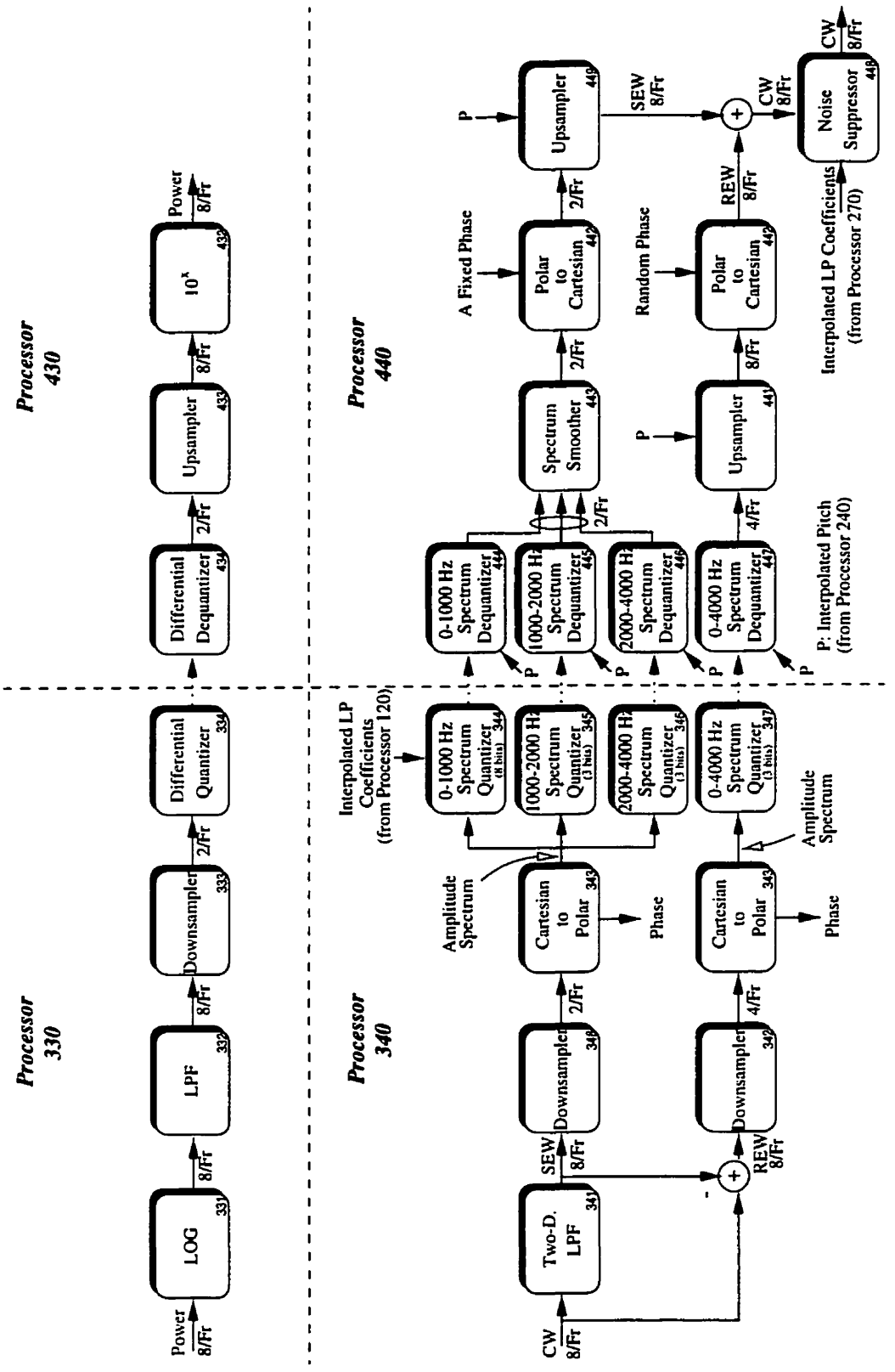


Fig. 3.8 The schematic diagrams of the quantizers and dequantizers for the power and the CW. The dotted arrows represent the bit-stream. (From [13].)

The effect of γ_w is to move the roots of $A(z)$ towards the origin , de-emphasizing the spectral peaks of $1/A(z)$. With γ_w as in Eq. (3.9), the response of $H_w(z)$ has valleys (anti-formants) at the formants locations and the inter-formant areas are emphasized. Thus noise is less audible if it shares the same spectral band. A typical value of γ_w is 0.8.

In our implementation we use the unquantized interpolated LP coefficients $\{a_k\}$ (in Eq. (3.9)) from processor **120**. The perceptually weighted error in processor **344** is obtained by multiplying the magnitude response $|H_w(z)|$ by the square of the difference between the original spectra and the corresponding codebook element.

As we will see in Section 5.2.1 the phase spectrum of the SEW is not important due to its very small evolution bandwidth. For this reason Choy's low bit-rate WI coder does not transmit any SEW phase information and at the receiver side a fixed phase spectrum is used.

REW Quantization

The REW contains the aperiodic component in the speech signal. Thus it contains little redundancy and requires a high bit-rate for accurate transmission. Again, it is found that for unvoiced speech the human hearing system observes only the power contour and the spectral power envelope [43]. This means that, the phase spectrum of the rapidly evolving part of the CW i.e., the REW does not need to be transmitted while its magnitude spectrum, that is to be transmitted, can be replaced by a signal with the same magnitude spectrum and a similar signal-power contour without a decrease in the perceived naturalness of the speech signal. Hence, REW magnitude spectrum does not require a high accuracy in its transmission and hence it is possible to reconstruct the surface $u_{REW}(n, \phi)$ in a perceptually accurate fashion at low bit rates.

Typically, the REW is downsampled to a rate of 200 Hz (four REWs per frame) as shown in Fig. 3.8. The amplitude spectrum of each downsampled REW is transformed into fixed dimensional polynomial coefficients and these coefficients are vector quantized using a codebook with only eight different spectral shapes. The phase spectrum of REW is not transmitted and is modelled at the decoder by using random phase for each frequency component.

3.3 WI Decoder

The decoder receives the quantized version of pitch, LSFs, gain and the SEW-REW magnitude spectrum. The block diagram shown in Fig. 3.9 depicts the decoding and synthesis stages. At the decoder, first both the SEWs and the REWs are upsampled to a rate of 400 Hz as shown in Fig. 3.8; the CWs are obtained by simply adding the two together. In an attempt to eliminate the coding noise introduced by the quantization of SEW and REW, the CWs are passed through a formant-based postfilter [4].

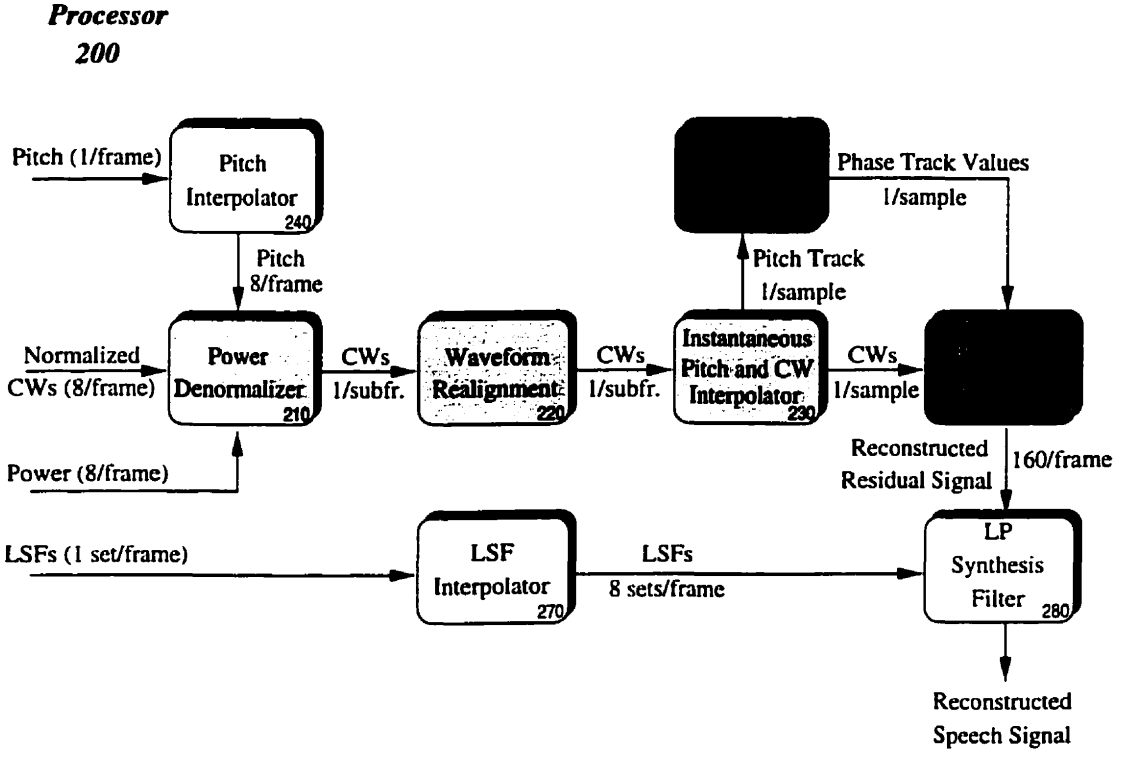


Fig. 3.9 A block diagram of the WI decoder. The update rates of the lighter-shaded, darker-shaded and non-shaded processors are once per subframe, once per sample, and once per frame respectively. (From [13].)

After power denormalization and subsequent realignment, the two-dimensional CW surfaces are converted back into the one-dimensional residual signal using a CW and a pitch length at every sample point obtained by linear interpolation. This conversion process, as is shown in Fig. 3.10, also requires the phase track estimated from the pitch value at each sample point.

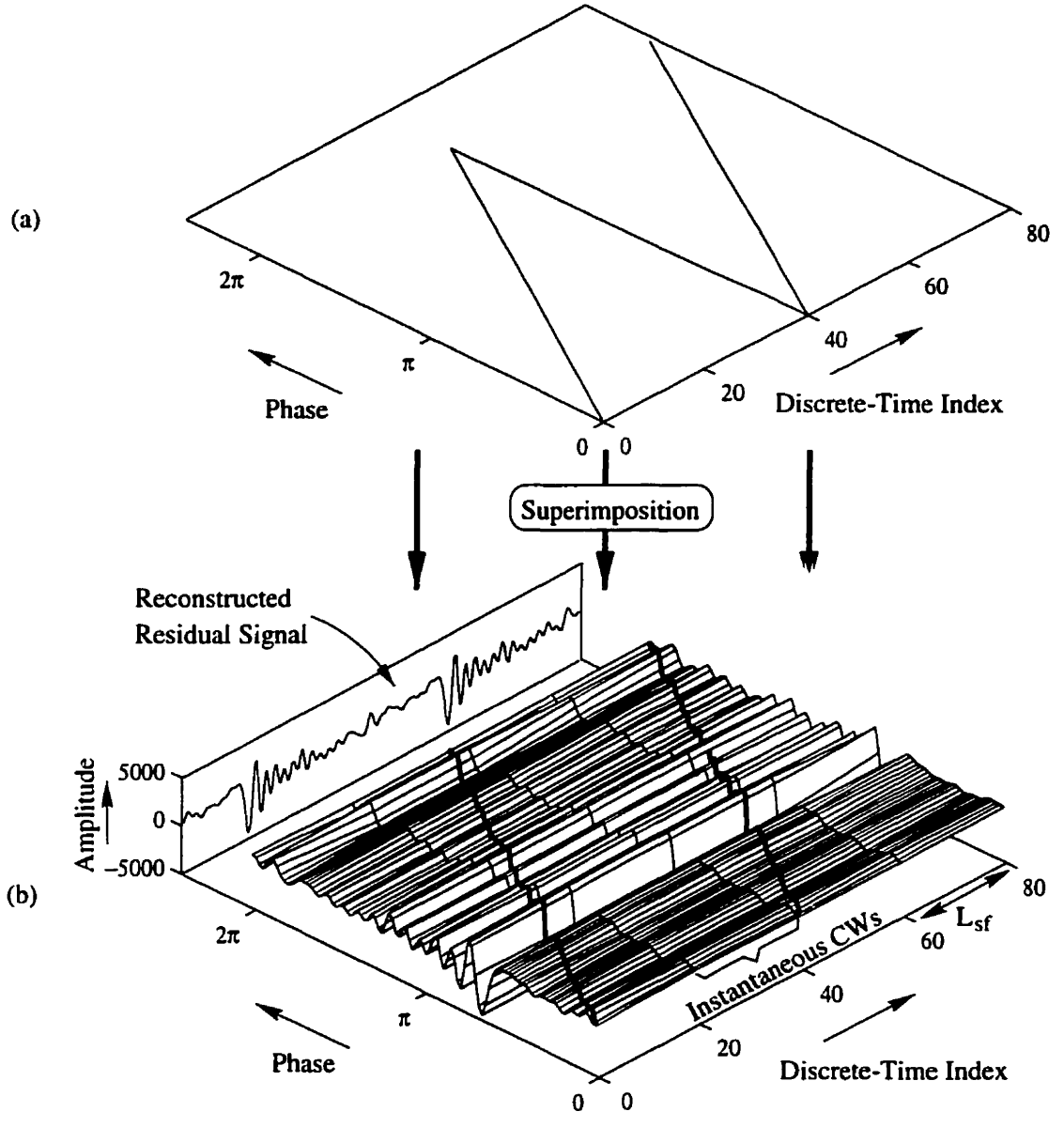


Fig. 3.10 Construction of the one dimensional residual signal from the reconstructed two dimensional CW surface using continuous interpolation. (a) An interpolated (instantaneous) phase track for a voiced segment which has a constant pitch at 40. (b) The interpolated pitch track superimposed on the interpolated CW surface. (From [13].)

The reconstructed 1-D residual signal is used to excite the LP synthesis filter to obtain the output speech signal. The software implementation of the different blocks of decoder exploits the following mathematical discrete-time equations:

- The phase track is computed by incrementally summing the area under the frequency track curve, which is reciprocal to the corresponding pitch value and is approximately given by:

$$\phi(n) \approx \phi(n-1) + \pi\left(\frac{1}{P(n-1)} + \frac{1}{P(n)}\right) \tag{3.10}$$

- Over the interpolation interval $n_i \leq n \leq n_{i+1}$, the reconstructed discrete-time one-dimensional residual signal is given by:

$$\hat{e}(n) = \sum_{k=1}^{\lfloor P(n)/2 \rfloor} \begin{aligned}&\{[(1-\alpha(n))A_k(n_i) + \alpha(n)A_k(n_{i+1})]\cos(k\phi(n)) \\ &+ [(1-\alpha(n))B_k(n_i) + \alpha(n)B_k(n_{i+1})]\sin(k\phi(n))\}\end{aligned} \quad , \; 0 \leq \phi(.) \leq 2\pi \tag{3.11}$$

A detailed discussion on Eq. (3.10) can be found in [13]. Eq. (3.11) is a modified inverse DTFS operation where the DTFS coefficients are linearly interpolated.

3.4 Summary

In this chapter we have reviewed the existing WI coder, originally implemented by Choy [13] based on Kleijn's frequency domain approach. Our primary concern is with the SEW coding scheme. The next chapter provides the guideline on how the SEW can be efficiently coded using a hybrid product code vector quantization technique, exploiting the inherent properties of the SEW discussed in this chapter. The issue of perceptually irrelevant phase information in speech coder will also be addressed in the following chapter.

Chapter 4

Efficient SEW Coding

In the waveform interpolation paradigm, the voiced part is modelled as a slowly evolving waveform (SEW); therefore, the quality of the waveform interpolation model depends largely on the efficient quantization of the SEW [37, 44]. In the first part of this chapter, we discuss the basics of vector quantization, followed by a novel technique we have used to efficiently quantize the amplitude spectrum of SEW. The second part deals with the phase information, whose scope is not only limited to the WI model, but can also be extended to any parametric coder.

4.1 Vector Quantization[1]

4.1.1 Introduction and Background

Vector quantization (VQ) is a lossy data compression method based on the principle of *source coding*, a terminology due originally to Shannon in his classic development of information theory, "A Mathematical Theory of Communication" [45, 46]. In this paper, he formulated the theory of data compression, in which he established that there is a fundamental limit to *lossless data compression*. This limit, called the *entropy rate*, is denoted by H. The exact value of H depends on the information source—more specifically, the statistical nature of the source. It is possible to compress the source, in a lossless manner, with compression rate close to H. Moreover, it is impossible to do better than H.

[1]The organization of this section follows from Prof. M. Kaplan's lectures on "Probability and Random Signals–II", Dept. of Electrical & Computer Engineering, McGill University, Canada.

Shannon also developed the theory of *lossy data compression*, known better as *rate-distortion theory*. In lossy data compression, the decompressed data does not have to be exactly the same as the original data. Instead, some amount of *distortion*, D, is tolerated. Shannon showed that, for a given source (with all its statistical properties known) and a given *distortion measure*, there is a function, $R(D)$, called the *rate-distortion function*. The theory says that if D is a tolerable amount of distortion, then $R(D)$ is the best possible compression rate. When a given input block or vector is encoded following this rate constraint on the code, the encoder must select the binary codeword which, when decoded, yields a reproduction with minimum distortion of the input with respect to all possible reproductions. Shannon viewed such a block code as a *source code subject to a fidelity criterion*, but a code of this type can also be called a *vector quantizer* and the operation of this code can be called *vector quantization* since it is a natural generalization of a simple quantizer to vectors. It is a mapping of a real vectors (an ordered set of signal samples) into binary vectors using a minimum distortion rule. Unlike Shannon, however, it is not required that the coders map consecutive blocks in an independent fashion. In other words, the encoder and decoder may have memory as do predictive codes and finite state codes. Lossless data compression theory and rate-distortion theory are known collectively as *source coding theory* which sets fundamental limits on the performance of all data compression algorithms.

Rate Distortion Theory

To formally present rate-distortion theory, we have to define the notion of distortion. Distortion in reproduction of a source is a measure of fidelity or closeness of the reproduction to the original source output. In a high-fidelity reproduction, the reproduced signal is very close to the original signal with low distortion. A *distortion measure* is a mathematical entity which specifies exactly how close the approximation is. Generally, it is a function which assigns to any two letters x and $\hat{x}$, in the alphabet $\mathcal{A}$, a non-negative number denoted as

$$d(x, \hat{x}) \geq 0, \tag{4.1}$$

where x is the original data, $\hat{x}$ is the approximation, and $d(x, \hat{x})$ is the amount of distortion between x and $\hat{x}$. Two common distortion measures are:

- *Hamming Distortion*: In the discrete case this is the most common distortion measure

and is defined by

$$d(x,\hat{x}) = \begin{cases} 0, & \text{if } x = \hat{x} \\ 1, & \text{if } x \neq \hat{x}. \end{cases} \tag{4.2}$$

- *Squared-Error Distortion*: In the continuous case this is the most frequently used distortion measure and is given by

$$d(x,\hat{x}) = (x-\hat{x})^2. \tag{4.3}$$

It is also assumed that we are dealing with a *per letter distortion measure*, meaning that the distortion between sequences is the average of the distortion between their components, i.e.,

$$d(\mathbf{x}^n,\hat{\mathbf{x}}^n) = \frac{1}{n}\sum_{i=1}^{n} d(x_i,\hat{x}_i). \tag{4.4}$$

This assumption simply means that the position of the "error" in reproduction is not important and the distortion is not context dependent. With the help of the above-stated notion of distortion, let us now discuss rate-distortion theory. Rate-distortion theory states that for a given source and distortion measure, there exists a function, $R(D)$, called the rate-distortion function. The typical shape of $R(D)$ is shown in Fig. 4.1.

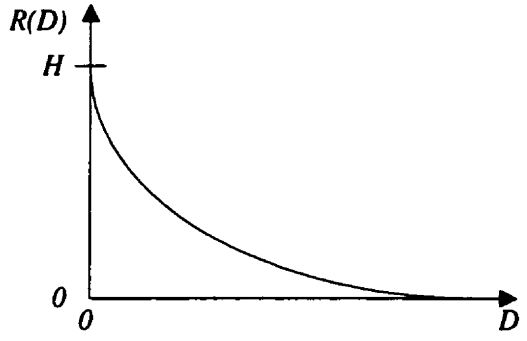


Fig. 4.1 Rate-Distortion function.

If the source samples are independent of one another, the rate-distortion function can be obtained by solving the constrained minimization problem:

$$R(D) = \min_{Q_{j|i}} \sum_{i=1}^{m} p_i \sum_{j=1}^{m} Q_{j|i} \log_2 \left[\frac{Q_{j|i}}{\sum_{k=1}^{m} p_k Q_{j|k}} \right] \tag{4.5}$$

subject to the constraints

$$\begin{aligned} & 0 \leq Q_{j|i} \leq 1, \\ & \sum_{j=1}^{m} Q_{j|i} = 1, \quad \text{and} \\ & \sum_{i=1}^{m} p_i \sum_{j=1}^{m} Q_{j|i} d(i,j) \leq D \end{aligned} \tag{4.6}$$

where $d(i,j)$ is the distortion between the i-th and j-th letter in the alphabet. Using *Blahut's algorithm* [47] it may be possible to numerically calculate the rate-distortion function.

Rate-distortion theory is based on the concept of block coding (similar to above). A lossy block code is known as a vector quantizer. There does not exist a VQ with distortion D and rate less than $R(D)$. The block length n of the code is known as the VQ dimension.

Vector quantization is a generalization of a scalar quantization to the quantization of a vector. So before going to the details of VQ we would like to present a short description of scalar quantization scheme in the following subsection.

4.1.2 Scalar Quantization

In general, each source output is a real number, but transmission of real numbers requires an infinite number of bits. Therefore, it is required to map the set of real numbers into a finite set and simultaneously minimize the distortion introduced. In scalar quantization, the set of real numbers $\mathcal{R}$ is partitioned into N disjoint subsets (known as *cells* or *Voronoi* regions) denoted by $\mathcal{R}_k$, $k = 1, 2, \ldots, N$. The cells take the form $\mathcal{R}_k = (a_{k-1}, a_k]$ where the a_k's, which are called *thresholds*, form an increasing sequence i.e., $a_1 < a_2 < \ldots < a_{N-1}$. Corresponding to each cell $\mathcal{R}_k$, a *representation point* (sometimes referred to as *output levels*) $\hat{x}_k$, which usually belongs to $\mathcal{R}_k$, is chosen. If the source output at time i, x_i, belongs to $\mathcal{R}_k$, then it is represented by $\hat{x}_k$, which is the quantized version of x_i. $\hat{x}_k$ is then represented by a binary sequence and transmitted. The quantization function is then

defined by

$$Q(x) = \hat{x}_k \quad \text{for all } x \in \mathcal{R}_k. \tag{4.7}$$

Due to the non-invertible nature of the quantization function some information is lost in the process of quantization. If we are using the squared error distortion measure, then

$$d(x, \hat{x}) = (x - Q(x))^2 = e^2. \tag{4.8}$$

In practice, the average distortion is considered as a measure of the quality of a system, with smaller average distortion meaning higher quality. Since X is a random variable, so are $\hat{X}$ and e; therefore, the average distortion, also known as quantization noise becomes an expectation, namely

$$D = E[d(X, \hat{X})] = E[X - Q(X)]^2 = E[X - \hat{X}]^2 = \int_{-\infty}^{\infty} (x - Q(x))^2 f_X(x) dx, \tag{4.9}$$

where $f_X(x)$ is the pdf of X. For an N-level scalar quantizer, the N regions are defined as, $\mathcal{R}_1 = (-\infty, a_1]$, $\mathcal{R}_2 = (a_1, a_2], \ldots, \mathcal{R}_N = (a_{N-1}, +\infty)$ and thus Eq. (4.9) becomes

$$D = \sum_{k=1}^{N} \int_{\mathcal{R}_k} (x - Q(x))^2 f_X(x) dx. \tag{4.10}$$

The quantization process can be modelled as in Fig. 4.2.

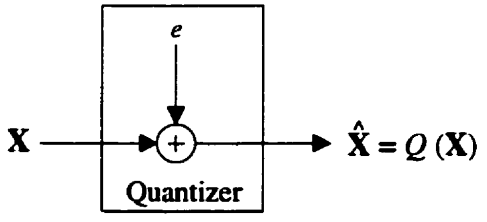


Fig. 4.2 Additive noise model of a quantizer.

Depending upon the length of the quantization region the scalar quantizer can be divided into two classes: uniform and nonuniform quantizers.

Uniform quantizers are the simplest of scalar quantizers. In an N-level uniform quantizer, all regions except the outermost cells, $\mathcal{R}_1$ and $\mathcal{R}_N$ are of equal length, which is denoted by the step size Δ and the thresholds a_i are midway between adjacent levels. Thus

for $1 \leq i \leq N-2$, we have $a_{i+1} - a_i = \Delta$. This quantization scheme is matched to uniform probability distribution functions. In a uniform quantizer, the distortion is given by

$$\begin{aligned} D &= \sum_{k=1}^{N} \int_{\mathcal{R}_k} (x - \hat{x}_k)^2 f_X(x) dx \\ &= \int_{-\infty}^{a_1} (x - \hat{x}_1)^2 f_X(x) dx + \sum_{i=1}^{N-2} \int_{a_1+(i-1)\Delta}^{a_1+i\Delta} (x - \hat{x}_{i+1})^2 f_X(x) dx \\ &\quad + \int_{a_1+(N-1)\Delta}^{\infty} (x - \hat{x}_N)^2 f_X(x) dx. \end{aligned} \tag{4.11}$$

Fig. 4.3 is an example of an eight level uniform quantizer.

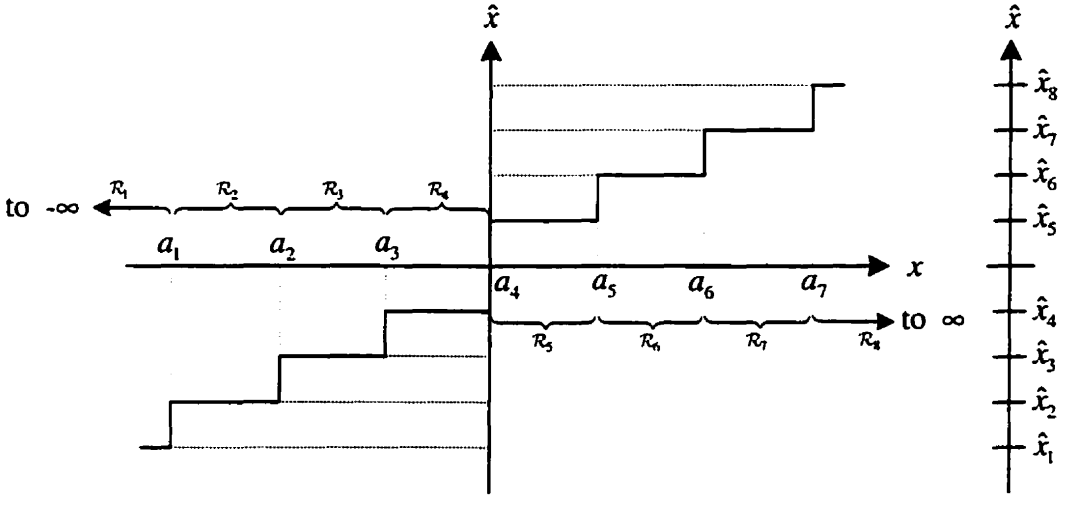


Fig. 4.3 Example of an 8-level quantization scheme.

In *nonuniform scalar quantizer* the quantization regions need not necessarily be equal. There are two major advantages to using nonuniform spacing of quantization levels.

- It is possible to significantly increase the dynamic range that can be accommodated for a given number of bits of resolution by using a suitably chosen nonuniform quantizer.
- It is possible to design a quantizer tailored to the specific input statistics so that considerably superior SNR is attained for a given resolution and given input pdf when the levels are allowed to be non-uniformly spaced.

Some popular schemes of nonuniform quantization are μ-law and A-law methods which are used to quantize speech signals [48]. In these quantization schemes, the nonlinear operation is a piecewise approximation to a logarithmic function. Given a uniform quantizer with cell width Δ, the region of the input space within $\Delta/2$ of some quantizer level is called the *granular region* or simply the *support* and that outside (where the quantizer error is unbounded) is called the *overload* or *saturation* region. More generally, the support or granular region of a nonuniform quantizer is the region of the input space within a relatively small distance of some level, and the overload region is the complement of the granular region.

Design of Optimal Scalar Quantizer

In the nonuniform quantizer, we relax the condition that the quantization regions be of equal length. As a result, we minimize the distortion with fewer constraints, resulting in a nonuniform quantizer which may perform better relative to a uniform quantizer with the same number of levels. The design problem of an optimal scalar quantizer now comes down to find a total of $2N-1$ variables, $\{a_i\}$ for $1 \leq i \leq N-1$ and $\{\hat{x}_i\}$ for $1 \leq i \leq N$, that minimizes the distortion measure given by

$$\begin{aligned} D &= \sum_{k=1}^{N} \int_{\mathcal{R}_k} (x-\hat{x}_k)^2 f_X(x)dx \\ &= \int_{-\infty}^{a_1} (x-\hat{x}_1)^2 f_X(x)dx + \sum_{i=1}^{N-2} \int_{a_i}^{a_{i+1}} (x-\hat{x}_{i+1})^2 f_X(x)dx \\ &\quad + \int_{a_{N-1}}^{\infty} (x-\hat{x}_N)^2 f_X(x)dx. \end{aligned} \tag{4.12}$$

Differentiating with respect to a_i yields

$$\frac{\partial}{\partial a_i} D = f_X(a_i)[(a_i-\hat{x}_i)^2 - (a_i-\hat{x}_{i+1})^2] = 0. \tag{4.13}$$

The solution of this equation is

$$a_i = \frac{1}{2}(\hat{x}_i + \hat{x}_{i+1}). \tag{4.14}$$

This result means that, in an optimal quantizer, *the boundaries of the quantization regions are the midpoints of the quantized values.*

The quantized values $\hat{x}_i$, is calculated by differentiating D with respect to $\hat{x}_i$ and define $a_0 = -\infty$ and $a_N = +\infty$. Thus, we obtain

$$\frac{\partial}{\partial \hat{x}_i} D = \int_{a_{i-1}}^{a_i} 2(x - \hat{x}_i) f_X(x) dx = 0. \tag{4.15}$$

The solution of this equation

$$\begin{aligned} \hat{x}_i &= \frac{\int_{a_{i-1}}^{a_i} x f_X(x) dx}{\int_{a_{i-1}}^{a_i} f_X(x) dx} = \int_{\mathcal{R}_i} x f_{X|\mathcal{R}_i}(x) dx \\ &= \frac{\int_{a_{i-1}}^{a_i} x f_X(x) dx}{p(a_{i-1} < X \leq a_i)} \\ &= \int_{a_{i-1}}^{a_i} x \frac{f_X(x)}{p(a_{i-1} < X \leq a_i)} dx \\ &= \int_{-\infty}^{+\infty} x f_X(x | a_{i-1} < X \leq a_i) dx = E[X | a_{i-1} < X \leq a_i]. \end{aligned} \tag{4.16}$$

It is immediately obvious from the above result that the value of $\hat{x}_i$ that minimizes distortion is the centroid $E[X|a_{i-1} < X \leq a_i]$ of the conditional pdf of X given that X lies in $\mathcal{R}_i$, $f_{X|\mathcal{R}_i}(x)$. Eq. (4.14) and Eq. (4.16) give the necessary conditions for a scalar quantizer to be optimal and are known as the *Lloyd-Max conditions*. In summary, an optimal quantizer must satisfy the following *optimality conditions*:

- The boundaries of the quantization regions are the midpoints of the corresponding quantized values (*nearest neighbor condition*).
- The quantized values are the centroids of that part of the input pdf that lies in the quantization regions (*centroid condition*).

There are no closed-form solutions to the problem of optimal quantization for general distributions. The repeated application of the improvement step, however, yields an *iterative* algorithm which attempts to reduce the average distortion at each iteration. The usual

method of designing the optimal quantizer using iterative method is to begin with a set of quantization regions and then find the quantized values using the centroid condition. The nearest neighbor condition is then applied to design the new quantization regions for the new quantized values. This process is repeated between the two steps until the distortion does not change appreciably from one step to the next.

4.1.3 Vector Quantizer Design

The extension of scalar quantization to higher dimensions leads to vector quantization. The idea of vector quantization is to take blocks of source outputs (vector) of length n and map them into a finite set $\mathcal{C}$ (in n-dimensional Euclidean space) containing K output or reproduction points, called *code vectors* or *codewords*. Thus,

$$Q : \mathcal{R}^n \rightarrow \mathcal{C}, \tag{4.17}$$

where $\mathcal{C} = (\hat{\mathbf{x}}_1, \hat{\mathbf{x}}_2, \ldots, \hat{\mathbf{x}}_K)$ and $\hat{\mathbf{x}}_p = (x_{p,1}, x_{p,2}, \ldots, x_{p,n}) \in \mathcal{R}^n$ for each $p \in \mathcal{J} \equiv \{1, 2, ..., K\}$. The set $\mathcal{C}$ is called the *codebook* and has size K, meaning it has K distinct elements.

Let us assume that the quantization regions (i.e., cells or Voronoi regions) in the n-dimensional space are denoted by $\mathcal{R}_i$, $i = 1, 2, \ldots, K$. These K regions partition the n-dimensional space. A cell that is *unbounded* is called an *overload cell* and the collection of all overload cells is called the *overload region*. A bounded cell, i.e., one having finite (n-dimensional) volume, is called a *granular cell*. The collection of all granular cells is called the *granular region*.

A vector quantizer can be decomposed into two component operations, the vector *encoder* and the vector *decoder*. The encoder $\mathcal{E}$ is the mapping from $\mathcal{R}^n$ to the index set $\mathcal{J}$, and the decoder $\mathcal{D}$ maps the index set $\mathcal{J}$ into the reproduction set (codebook) $\mathcal{C}$. Thus,

$$\mathcal{E} : \mathcal{R}^n \rightarrow \mathcal{J} \quad \text{and} \quad \mathcal{D} : \mathcal{J} \rightarrow \mathcal{R}^n. \tag{4.18}$$

The overall partition of VQ can be regarded as the cascade or composition of two operations:

$$Q(\mathbf{x}) = \mathcal{D}.\mathcal{E}(\mathbf{x}) = \mathcal{D}(\mathcal{E}(\mathbf{x})). \tag{4.19}$$

Fig. 4.4 illustrates how the cascade of an encoder and decoder defines a quantizer. In the

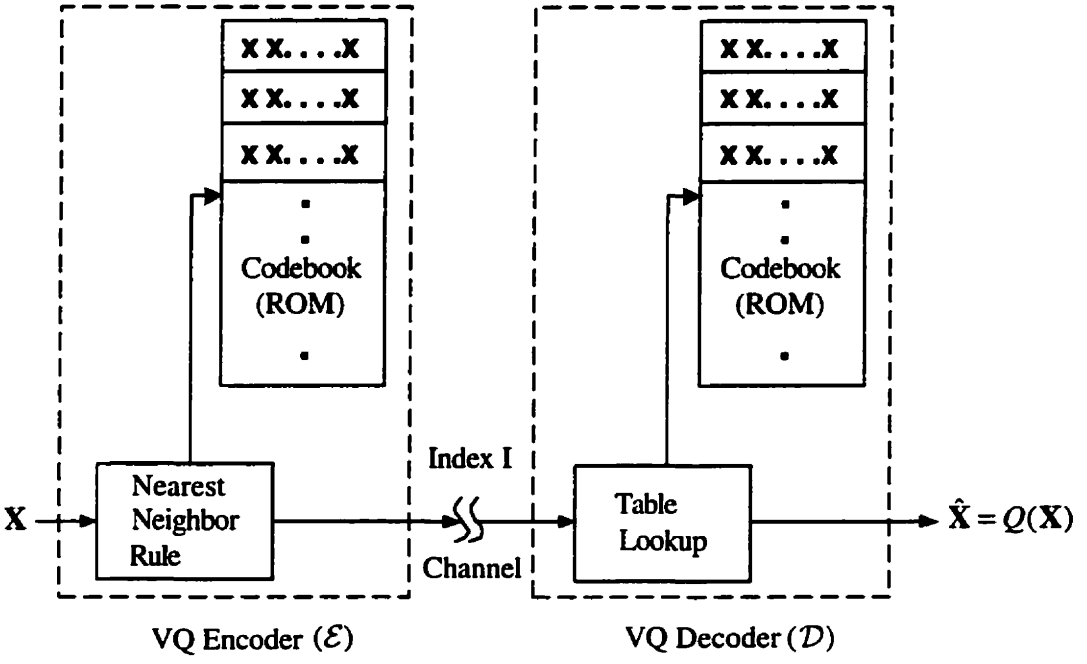


Fig. 4.4 Block diagram of a vector quantizer represented as the cascade of an encoder and decoder.

context of a digital communication system, the encoder of a vector quantizer encodes each input vector by finding its nearest neighbor from the codebook; the index of the best match is then sent (as a binary word) to the decoder. The decoder is a simple look-up table that uses the index to produce the reconstructed vector.

VQ Design: Analytical method

The VQ design problem can be stated as follows: given a vector source with known statistical properties, a distortion measure, and the number of codevectors along with its dimension, find a codebook and a partition of the cells which result in the smallest average distortion.

Let us assume that each block of source output of length n is denoted by $\mathbf{x} \in \mathbb{R}^n$. As in the previous section, let the K-codevectors representing the codebook are

$$\mathcal{C} = (\hat{\mathbf{x}}_1, \hat{\mathbf{x}}_2, \ldots, \hat{\mathbf{x}}_K). \tag{4.20}$$

Each codevector is n-dimensional and is given by

$$\hat{\mathbf{x}}_i = (\hat{x}_{i,1}, \hat{x}_{i,2}, \ldots, \hat{x}_{i,n}) \in \mathcal{R}^n, \quad \text{for } i = 1, 2, \ldots, K. \tag{4.21}$$

Let $\mathcal{R}_i$ be the Voronoi region associated with codevector $\hat{\mathbf{x}}_i$. Thus, the partition of the n-dimensional space is denoted by

$$\mathcal{V} = (\mathcal{R}_1, \mathcal{R}_2, ..., \mathcal{R}_K). \tag{4.22}$$

If the source vector $\mathbf{x}$ is in the encoding region $\mathcal{R}_i$, then it is quantized to

$$Q(\mathbf{x}) = \hat{\mathbf{x}}_i, \quad \text{if } \mathbf{x} \in \mathcal{R}_i. \tag{4.23}$$

The optimal vector quantizer of dimension n and number of levels K is the one that chooses the region $\mathcal{R}_i$'s and the quantized values $\hat{\mathbf{x}}_i$'s such that the resulting distortion is minimized, assuming a squared-error distortion measure. Applying the same procedure that we used for the case of scalar quantization, we obtain the following criteria for an optimal vector quantizer design:

- *Nearest neighbor condition*: Region $\mathcal{R}_i$ is the set of all points in the n-dimensional space that are closer to $\hat{\mathbf{x}}_i$ than any other $\hat{\mathbf{x}}_j$, for all $j \neq i$.

$$\mathcal{R}_i = \{\mathbf{x} \in \mathcal{R}^n : \|\mathbf{x} - \hat{\mathbf{x}}_i\| < \|\mathbf{x} - \hat{\mathbf{x}}_j\|, \forall j \neq i\}. \tag{4.24}$$

  For those vectors lying on the boundary, any tie-breaking procedure is satisfactory.

- *Centroid condition*: $\hat{\mathbf{x}}_i$ is the centroid of the region $\mathcal{R}_i$.

$$\hat{\mathbf{x}}_i = \frac{1}{p(\mathbf{X} \in \mathcal{R}_i)} \int\int \cdots \int_{\mathcal{R}_i} \mathbf{x} f_{\mathbf{X}}(\mathbf{x}) d\mathbf{x}. \tag{4.25}$$

These optimality conditions for the VQ design is the generalization of Lloyd's condition for designing optimal scalar quantizer. The analytical approach to designing optimal vector quantizer exploiting generalized Lloyd conditions is based on the same approach employed in designing the optimal scalar quantizers.

VQ Design: Practical Approach

An iterative algorithm is used to design a VQ codebook. This algorithm exploits the generalized Lloyd conditions for optimality of the iterative codebook modification operation, and hence it is referred to as the Generalized Lloyd Algorithm (GLA).

The computation of the centroids using centroid condition, by evaluating multiple integral, is generally impossible by analytical methods. However, an adequate analytical description of the input pdf is generally not available in most applications. Instead, a sample distribution based on empirical observations of the input vector, called the *training set*, is used to generate the improved codebook. In fact, this approach to be described next is equivalent to a Monte Carlo method for evaluating the needed integrals that determine the centroids. The GLA for VQ design is sometimes known as *k-means algorithm* after MacQueen [49] who studied it as a statistical clustering problem. Assume that there is a *training set* consisting of M source vectors:

$$\mathcal{T} = (\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_M). \tag{4.26}$$

Suppose M is sufficiently large such that the statistical properties of the source are captured by the training sequence. We assume that the source vectors are n-dimensional, i.e.,

$$\mathbf{x}_q = (x_{q,1}, x_{q,2}, \ldots, x_{q,n}) \in \mathcal{R}^n, \quad \text{for } q = 1, 2, \ldots, M. \tag{4.27}$$

Using the same notations for codewords, codebook, and Voronoi regions as have been used in the previous section, we write the optimality conditions (assuming a squared-error distortion measure) for discrete empirical data:

- *Nearest neighbor condition*:

$$\mathcal{R}_i = \{\mathbf{x} \in \mathcal{T} : \|\mathbf{x} - \hat{\mathbf{x}}_i\| < \|\mathbf{x} - \hat{\mathbf{x}}_j\|, \forall j \neq i\}. \tag{4.28}$$

- *Centroid condition*:

$$\hat{\mathbf{x}}_i = \frac{\sum_{\mathbf{x}_i \in \mathcal{R}_j} \mathbf{x}_i}{\sum_{\mathbf{x}_i \in \mathcal{R}_j} 1}, \quad i = 1, 2, \ldots, K. \tag{4.29}$$

This condition says that the codevector $\hat{\mathbf{x}}_i$ should be the average of all those training

vectors that are in encoding region $\mathcal{R}_j$. In implementation, one should ensure that at least one training vector belongs to each encoding region (so that the denominator in the above equation is never zero).

The Lloyd iteration can now be directly applied to the discrete input distribution defined from the training set $\mathcal{T}$ to obtain a locally optimal quantizer for this distribution. The actual design algorithm is stated concisely in Table 4.1. The flowchart for the GLA is

Table 4.1 Codebook design using the Generalized Lloyd Algorithm.

The Generalized Lloyd Algorithm
Step-1: Begin with an initial codebook $\mathcal{C}_1$. Set m = 1.
Step-2: Given the codebook, $\mathcal{C}_m$, perform the Lloyd iteration to generate the improved codebook $\mathcal{C}_{m+1}$. *Lloyd iteration*: *Step-2.1*: Given a codebook $\mathcal{C}_m = \{\hat{\mathbf{x}}_i\}$, partition the training set into cluster sets $\mathcal{R}_i$ using the Nearest Neighbor Condition. Use a suitable tie-breaking rule when necessary. *Step-2.2*: Using the Centroid Condition, compute the centroids for the cluster sets just found to obtain the new codebook, $\mathcal{C}_{m+1} = \{\text{cent}(\mathcal{R}_i)\}$. If an empty cell was generated in *Step-2.1*, an alternate codevector assignment is made (in place of the centroid computation) for that cell.
Step-3: Compute the average distortion for $\mathcal{C}_{m+1}$. If it has changed by a small enough amount since the last iteration, stop. Otherwise set $m + 1 \rightarrow m$ and go to *Step-2*.

shown in Fig. 4.5.

Fig. 4.6 shows the centroids and Voronoi (nearest neighbor) regions of a two-dimensional quantizer designed using the Lloyd algorithm on an independent and identically distributed (iid) sequence.

Since the design of a VQ system is a multidimensional optimization problem, there is a possibility that the codewords obtained may not be globally optimal [50]. Therefore, the initial codebook can have a great impact on the final codebook. Many methods have been proposed to mitigate this problem [50, 51]. Table 4.2 shows some of the initial codebook

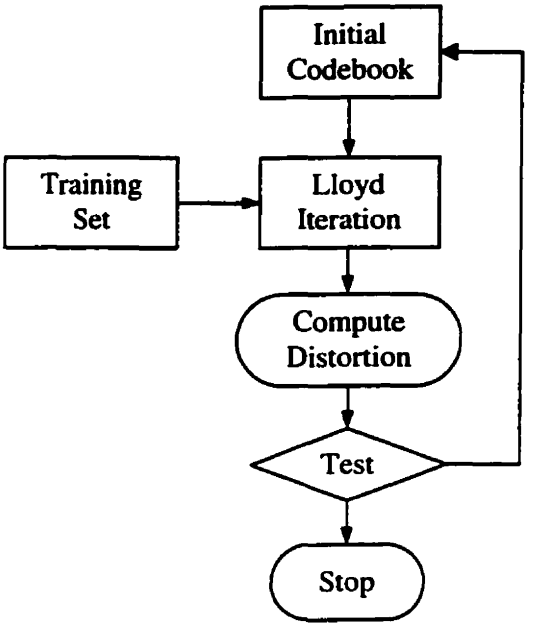


Fig. 4.5 Lloyd algorithm flow chart.

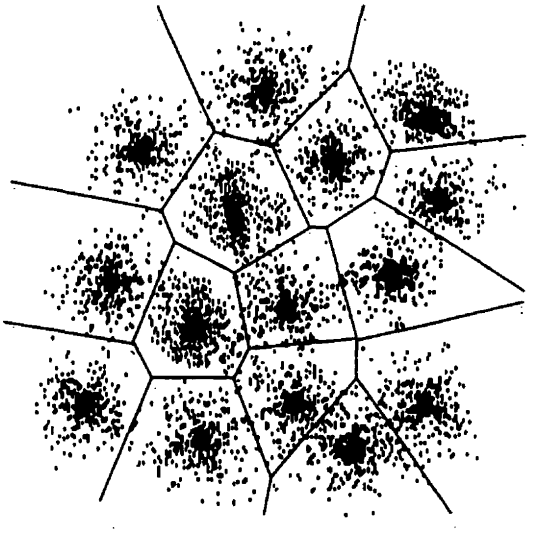


Fig. 4.6 Voronoi regions and centroids: two-dimensional iid source. In this example, the shaded points are called codevectors and the regions defined by the black borders are called Voronoi regions.

Table 4.2 VQ codebook design algorithms.

Methodology	Locally Optimal Techniques	Globally Optimal Techniques[1]
Conventional	Pruning [50], LBG (Splitting) [50], Pairwise Nearest Neighbor [53], Steepest Descent and Conjugate Gradient Methods [57].	Fuzzy C-Means [52], Fuzzy Vector Quantization [54], Simulated Annealing [55], [56], Deterministic Annealing [58].
Neural Network	Competitive Learning [59], Frequency Sensitive Competitive Learning [61], [62].	Competitive Learning and Soft Competition [60], Kohonen Self-Organizing Feature Maps [63], Fuzzy Kohonen Self-Organizing Feature Maps [64], Neural-Gas Network [65].

design techniques proposed in recent research literature. These techniques are categorized based on the methodology used to design the codebook and whether the algorithm seeks a local or a global minimum. GLA can be used on a codebook designed by any method given in Table 4.2 to further improve the performance of the codebook. However, in some cases, there may not be any improvement; that is, when the initial codebook is trapped in a local minimum or reached a global minimum, the subsequent application of GLA will not improve the performance of the codebook.

One of the widely used methods is the LBG (Linde-Buzo-Gray) procedure, which starts with creating a codebook with only one codeword. The first codeword is then split into two codewords to create the initial codebook to generate the second codebook. The iterative GLA method is used to find the final codebook at each step. This splitting and training process continue until the final codebook is obtained.

Evaluation of Vector Quantization

VQ is a potentially efficient representation of spectral information in the speech signal. In this subsection, we address both why and by how much a vector quantizer outperforms a scalar quantizer for the same source. Here we derive the advantages for constrained

[1]There is no mathematical proof for some of the techniques categorized as globally optimal that they search a globally optimal solution; however, they are generally independent of the initial codebook and, therefore, they have been categorized as globally optimal techniques.

resolution systems (i.e., systems having fixed number of output points) by exploiting the properties of optimal codebook design presented by Makhoul *et al* [66]. To facilitate the use of tractable equations for the performance of vector quantizers for any vector dimension, our discussion is restricted to only high-resolution quantization theory (we assume the rate, and hence the codebook size, is large). However, the tractability of high-resolution results makes them useful for understanding performance gains even in low rate cases.

To begin our discussion of "vector quantizer advantages" [67], let us first define the factors that represent the advantages of VQ over scalar quantization beginning with the expected distortion D produced by an n-dimensional vector quantizer with codebook size K

$$D(K;n) = C(n)K^{-2/n} \left\| p(x) \right\|_{n/(n+2)}, \tag{4.30}$$

where $C(n)$ is the coefficient of quantization for squared error distortion, $p(x)$ is the probability density function of the source vector $x = \{x_1, x_2, ..., x_n\}$, and the functional $\|.\|_\nu$ is defined as

$$\left\| p(x) \right\|_{n/(n+2)} = \left[\int p(x)^{n/(n+2)} dx \right]^{1+2/n} \tag{4.31}$$

This is the well-known Zador-Gersho [68] formula which gives the least distortion of any n-dimensional quantizer for high bit rates.

Gersho conjectured that the coefficient of quantization is determined by the moment of inertia of the optimal cell shape as

$$C(n) = \inf_{\mathcal{R}_n \in \hat{\mathcal{R}}_n} \left(\frac{1}{n} \right) \frac{\int_{\mathcal{R}_n} \|x - \hat{x}\|^2 dx}{[V(\mathcal{R}_n)]^{1+2/n}}, \tag{4.32}$$

where $\hat{x}$ is the centroid of the cell (considered to be a convex polytope) $\mathcal{R}_n$ belonging to $\hat{\mathcal{R}}_n$ that is the set of all admissible polytopes for n-dimensional space; For example, triangles, equilateral triangles, and hexagons are all admissible polytopes for $n = 2$, but the infimum in Eq. (4.32) is achieved when the hexagon is used in two dimension. For $n = 3$ the optimal polytope is the regular truncated octahedron [69]. Note that $V(\mathcal{R}_n)$ is the volume (Lebesgue integral) of the polytope $\mathcal{R}_n$.

To compare the distortion of the vector quantizer with that of scalar quantizer using a squared error criterion, Lookabaugh and Gray [67] expressed vector quantization gain as

the distortion ratio

$$\Delta(n) = \frac{D(K;1)}{D(K;n)} \tag{4.33}$$

By substituting Eq. (4.30) into Eq. (4.33) and rearranging them, we get

$$\Delta(n) = \underbrace{\frac{C(1)}{C(n)}}_{F(n)} \underbrace{\frac{\|\tilde{p}(x)\|_{1/3}}{\|p^*(x)\|_{n/(n+2)}}}_{S(n)} \underbrace{\frac{\|p^*(x)\|_{n/(n+2)}}{\|p(x)\|_{n/(n+2)}}}_{M(n)} \tag{4.34}$$

where $\tilde{p}(x)$ is the marginal density and $p^*(x)$ is defined as

$$p^*(x) = \prod_{i=0}^{n-1} \tilde{p}(x_i) \tag{4.35}$$

Here we have assumed that the source is stationary so that its marginal density does not depend on the coordinate.

The gain in Eq. (4.34) can now be decomposed into three factors, each of which represents a vector quantizer advantage. These factors are *space-filling advantage* $F(n)$, *shape advantage* $S(n)$, and *memory advantage* $M(n)$, and are defined in Eq. (4.34).

The Space-Filling Advantage

The definition of the space-filling advantage, $F(n)$, in Eq. (4.34) shows that it depends on the coefficients of quantization, and hence applying Gersho's conjecture (Eq. (4.32)), only on the efficiency with which polytopes can fill space.

We can evaluate $C(n)$ explicitly up to three dimensions only. For $n = 1, 2$, and 3, the optimal polytopes are the interval (there are no other convex polytopes in the one dimensional case), hexagon, and regular truncated octahedron, respectively. For other values of n, we must rely on bounds. Partitions based on Gersho's admissible polytopes are equivalent to lattices, where the points of a lattice are the centroids of the polytopes. Consequently, known lattices for various dimensions provide an upper bound on $C(n)$, and hence a lower bound on $F(n)$. Among these is the lattice formed by concatenating replicas of a uniform scalar quantizer, which, from Eq. (4.32), would yield a shape advantage of one. Since this is a lower bound, we clearly have that $F(n) \geq 1$ for $n \geq 1$.

Conway and Sloane [70] have studied lattices of various dimensionalities and have con-

jectured a lower bound on $C(n)$. Their work provides a conjectured upper bound on $F(n)$. It is obvious from their work (Fig. 4.7) that vector quantizers always outperform scalar quantizers.

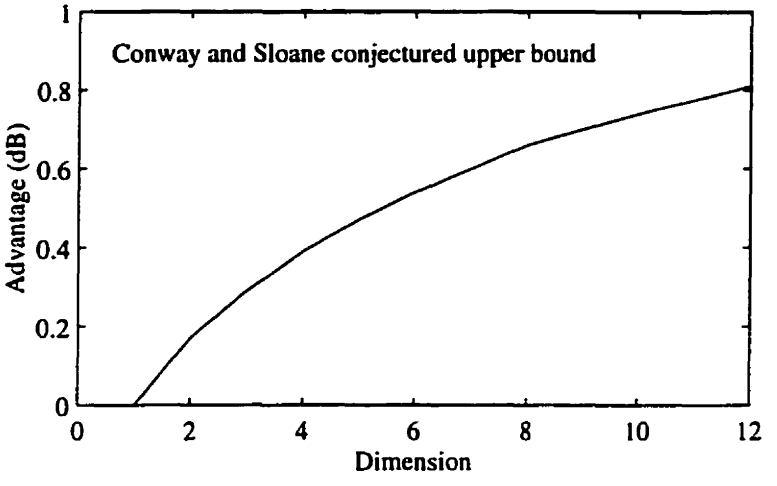


Fig. 4.7 Advantages of vector quantizer due to space filling advantage.

As $k \rightarrow \infty$, Conway and Sloane's conjectured upper bound has the limiting value

$$\lim_{k \rightarrow \infty} C(n) = (2\pi e)^{-1} \tag{4.36}$$

Thus, the maximum contribution by the space filling advantage is 1.43 dB.

The Shape Advantage

This advantage, $S(n)$, depends only on the shape of the marginal probability density function. VQ has a different performance for a different probability density function. Shape advantages for three different probability distribution functions—uniform, Gaussian and Laplacian densities are shown in Fig. 4.8. The maximum shape gain for Gaussian source with infinite dimension is 2.81 dB, while the gain for Laplacian source is 5.63 dB.

The Memory Advantage

This factor captures the non-linear characteristic of the source distribution. The value $M(n)$ depends on the correlation factor. If the vector components are totally independent and

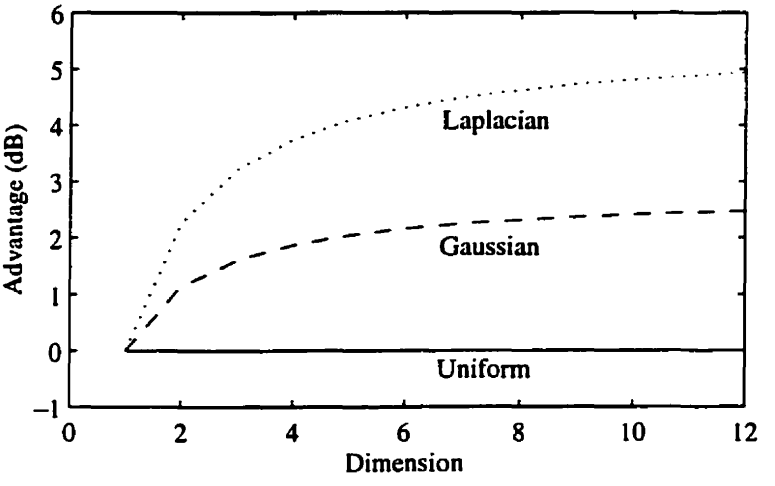


Fig. 4.8 Advantages of vector quantizer due to shape advantage.

identically distributed (iid), the ratio $M(n)$ is unity. The more dependent the components of the vector, the larger is the value of $M(n)$.

While it enjoys certain advantages, VQ has its heavy share of disadvantages too. Most of these stem from the need to use a codebook. The "Vector Quantizer Disadvantages" become more and more imposing as the vector dimension increases or the size of the codebook increases.

- The encoding process can be computationally intensive and slow. For instance, the complexity increases exponentially with the vector dimension. On the other hand, a performance drop will occur with small vector dimensions since the time-varying nature of the speech parameters cannot be taken into account and the bit rate will be high.

- Memory storage requirement for the codebook also increases rapidly with increasing vector dimension and codebook size. For real time VLSI implementation, this is also limited by the cost and size of ROM (Read Only Memory) modules.

- Codebook generation is a very lengthy process, especially with large vector dimensions and codebooks. However, since codebook generation is done offline, this disadvantage is somewhat allayed.

- As with many other compression schemes that divide the speech into subbands, the

blocking effect is apparent in low bit rate VQ. This is a disturbing artifact that can be seen as perceptible discontinuities across block boundaries.

Product Code VQ

The distortion performance of GLA for structured vector quantizers (VQ) is adequate, but its creation, storage and encoding complexities each grow exponentially in both dimension and rate. The complexities can be reduced by decomposing or partitioning vectors of high dimension into subvectors (sometimes called *feature vectors*), each of low dimensionality. Instead of a single VQ for the entire vector, each subvector can be separately encoded with its own codebook. By sending a set of indices to the decoder, the decoder reconstructs the original vector by first decoding each subvector and then concatenating these vectors to regenerate the reconstructed approximation to the original large vector. This is the general idea and objective of the product quantizer [71, 72]. The complexities of a product quantizer are the sums of those of the component quantizers.

Given the original input vector $\mathbf{X}$ of dimension $n > 1$, let $\mathbf{U}_1, \mathbf{U}_2, ... \mathbf{U}_\nu$ be a set of feature vectors that are function of $\mathbf{X}$ and jointly determine $\mathbf{X}$. If we denote the feature extraction functions as f_i for $i = 1, 2, ..., \nu$ and the synthesis function as g, then

$$\begin{aligned} \mathbf{U}_i &= f_i(\mathbf{X}), \quad \text{for } i = 1, 2, \ldots, \nu. \\ \mathbf{X} &= g(\mathbf{U}_1, \mathbf{U}_2, \ldots, \mathbf{U}_\nu). \end{aligned} \tag{4.37}$$

Each feature vector should be easier to quantize than $\mathbf{X}$ because it takes on values in a more compact region of n-dimensional space or has a lower dimensionality. For each i, let $\mathcal{C}_i$ be a codebook of N_i reproduction code vectors that contain the reproduction values for $\mathbf{U}_i$, which is $\hat{\mathbf{U}}_i \in \mathcal{C}_i$ for $i = 1, 2, \ldots, \nu$. Fig. 4.9 shows such a product code vector quantizer with independent VQ for each feature vector. As we shall illustrate in shape-gain VQ, this structure is not necessarily the optimal encoding configuration for product code VQ. There are a variety of special cases of product VQ. Two examples will be discussed hereafter.

Split VQ

The Split VQ structure divides a high dimensional vector into two or more subvectors of lower dimensions which are then independently vector quantized subject to the monotonic-

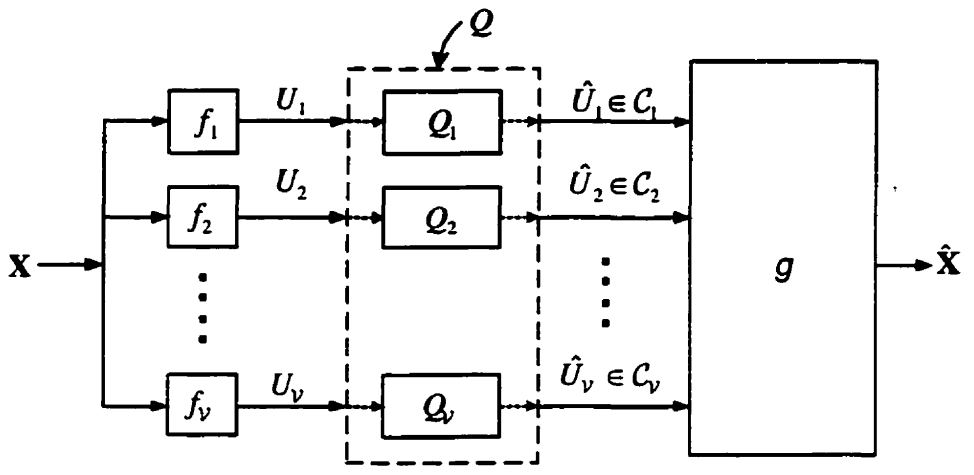


Fig. 4.9 Structure of product code VQ. Block Q corresponds to the general configuration. When block Q is substituted by blocks $Q_1, Q_2, \ldots, Q_\nu$ it becomes a product VQ with independent quantizers which is simpler to implement at the cost of optimality

ity property. An n-dimensional feature vector $\mathbf{X}$ is a concatenation of m subvectors $\mathbf{U}_i$ whose dimensions n_i sum up to n:

$$\mathbf{X} = [\mathbf{U}_1^T, \mathbf{U}_2^T, ..., \mathbf{U}_m^T]^T, \tag{4.38}$$

where

$$\sum_{i=1}^{m} n_i = n. \tag{4.39}$$

Hence, split VQ is also known as partitioned VQ, or concatenation product code VQ (CPC). This way the complexity is reduced at the expense of possibly a higher distortion.

Paliwal and Atal, in [42] showed that for a 10 dimensional LP coefficient vector a two way or three way splitting of LSF vectors gives reasonable improvement in performance (relative to scalar quantization) for LSF based VQ. Splitting the LSF vector corresponds to splitting the LPC power spectrum. Usually one would assign more bits to the lower frequency spectrum than the higher one because of the sensitivity of the ear to lower frequencies. Table 4.3 shows a comparative view of two-way split VQ (2-SVQ) and three-way split VQ (3-SVQ) in terms of their bit allocation scheme and the corresponding spectral distortion [73]. Obviously, full-search VQ offers better performance at the cost of significantly higher computational complexity (codebook size 2^{24}).

Table 4.3 Average spectral distortion (SD) of 2-SVQ and 3-SVQ for a 24 bit codebook when quantizing LSF vectors. (From [74].)

Scheme	Splitting (10-LSFs)	Bits Alloc. (24 bits/fr.)	Av. SD (dB)	SD Outliers (in %)	
				2–4 dB	>4 dB
2-SVQ	4, 6	12, 12	1.21	6.21	0.02
3-SVQ	3, 3, 4	8, 8, 8	1.29	7.0	0.05

The use of Paliwal weighted LSF distortion measure [4] further improves the VQ. The 3-SVQ approach, along with weighted LSF distortion measure, is used in our WI coder.

Split VQ technique is the most efficient scheme (in the sense of distortion-rate) if used with an adaptive bit allocation scheme in which the available bits are allocated to each subvector based on the local statistics. In our WI coder, we use a split VQ scheme along with a perceptually based bit allocation strategy.

Shape-Gain VQ

A *shape-gain vector quantizer* decomposes a source vector $\mathbf{X}$ into a scalar *gain* $g = \|\mathbf{X}\|$ and *shape* $\mathbf{S} = \mathbf{X}/g$, which are quantized to $\hat{g}$ and $\hat{\mathbf{S}}$, respectively, and the output is $\hat{X} = \hat{g}\hat{\mathbf{S}}$ (see Fig. 4.10). According to common practice, we assume the quantized shape satisfies $\left\|\hat{\mathbf{S}}\right\| = 1$. In this product code decomposition, the shape vector lies on the surface of a hypersphere in n-dimensional space and is therefore easier to quantize (with lower rate shape codebook) than the original vector $\mathbf{X}$. We assume that the same pattern of variations in $\mathbf{X}$ recurs with a wide variety of gain values, which suggests that the probability distribution of the shape is approximately independent of the gain. This allows the gain and shape quantizers to operate in parallel and independently of each other with very little compromise in optimality, especially for high resolution. An advantage of shape-gain VQ is that the encoding and storage complexities grow with the sum of the gain codebook size and shape codebook size, while the effective codebook size is the product of these quantities.

As with shape-gain VQ, the optimal lossy encoder will in general not view only one coordinate at a time. Separate and independent quantization of the components provide a low-complexity but generally suboptimal encoder. In order to determine the *optimal encoding structure* we start with the product reproduction codebook consisting of a gain codebook

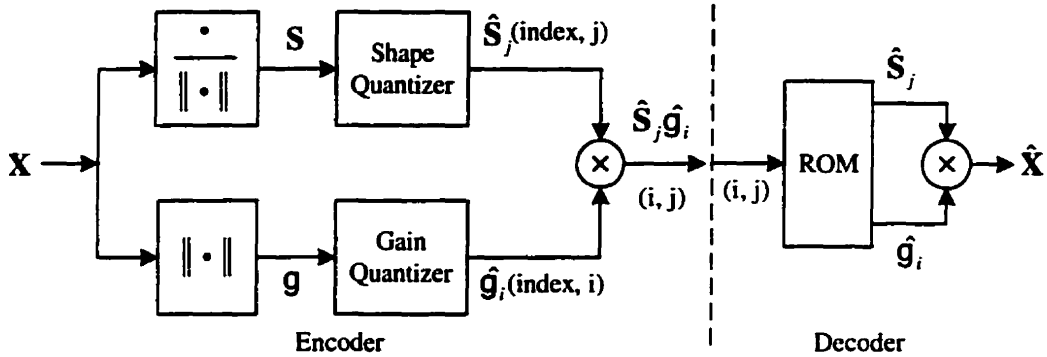


Fig. 4.10 Block diagram of independent shape-gain VQ.

$\mathcal{C}_g = \{\hat{g}_i;\ i = 1, 2, ..., N_g\}$ of positive scalars and a shape codebook $\mathcal{C}_s = \left\{\hat{\mathbf{S}}_j;\ j = 1, 2, ..., N_s\right\}$ of unit norm n-dimensional vectors. Assuming a squared error distortion measure, we have-

$$\begin{aligned} d(\mathbf{X}, \hat{g}\hat{\mathbf{S}}) &= \left\|\mathbf{X} - \hat{g}\hat{\mathbf{S}}\right\|^2 \\ &= \|\mathbf{X}\|^2 + \hat{g}^2 - 2\hat{g}(\mathbf{X}^t\hat{\mathbf{S}}). \end{aligned} \tag{4.40}$$

The above expression suggests [75] that the minimum-squared-error reproduction codeword $\hat{g}_i\hat{\mathbf{S}}_j$ for an input vector $\mathbf{X}$ can be found by the following algorithm:

- *Step-1*: Choose the index j that maximizes the correlation $\mathbf{X}^t\hat{\mathbf{S}}_j$.
- *Step-2*: For the chosen j choose the index i minimizing $\left|\hat{g}_i - \mathbf{X}^t\hat{\mathbf{S}}_j\right|$.

This sequential rule gives the minimum-squared-error reproduction codeword without explicitly normalizing the input vector (which would be computationally expensive). The encoding algorithm is depicted in Fig. 4.11.

As an example, shape-gain VQs and full search VQs were designed [71] for the same speech data set. For each resolution and dimension various sizes of shape and gain codebooks were tried. The detailed design algorithm is described in [71]. The results for the best choices of the gain and shape codebook sizes are summarized in Fig. 4.12. As expected, for a common resolution and dimension, shape-gain VQ is inferior to full search VQ in terms of performance, but it has the reduced complexity of a product code. Because of the reduced complexity, shape-gain VQ is capable of being used at higher dimensions than full search VQ.

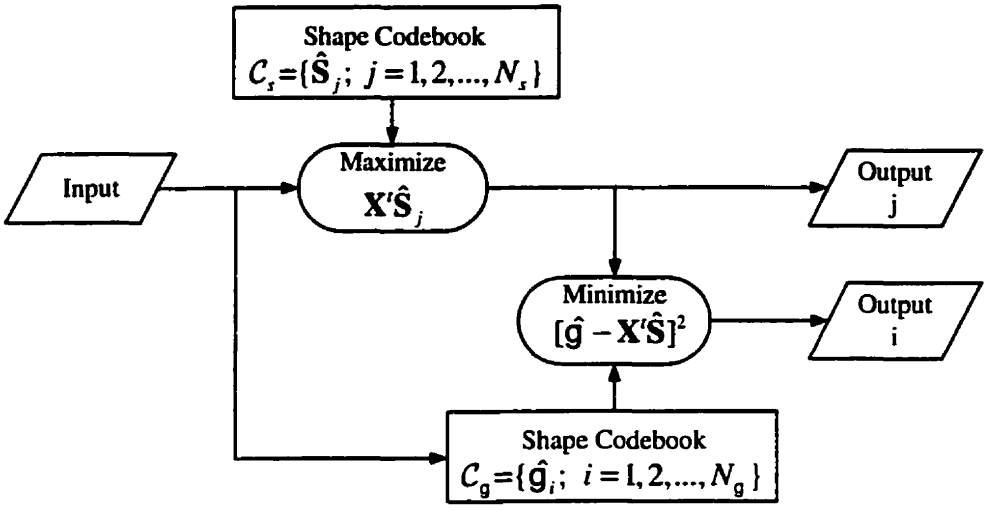


Fig. 4.11 Shape-Gain VQ encoder flow chart.

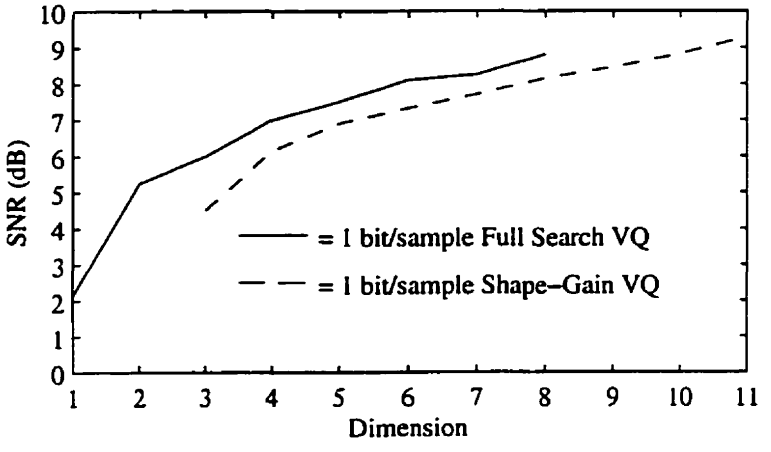


Fig. 4.12 SNR for shape-gain VQ of sampled speech. (From [50].)

4.1.4 Particular Distortion Measure of Interest

In speech processing, mean squared error (MSE) is the most commonly used distortion measure for evaluating the performance of compression algorithms because of rich theory and ease of use. In particular, for quantization or source coding, it is simpler to design efficient encoders and decoders which use mean squared error distortion criterion. It has often been empirically shown, however, that mean squared error does not correlate well with subjective (human) quality assessments [26]. As a result, decreasing the mean squared error does not necessarily improve speech quality. As standards for speech quality become more demanding, code designers require distortion measures which are more consistent with human perception of speech. As a result, perceptual distortion measures are receiving more attention.

The most popular perceptual distortion measure is the perceptually weighted mean squared error [2]. This measure takes the human ear's nonlinear perception of speech into account by allowing more emphasis on the low frequency spectrum of the signal to which human ear is most sensitive. The perceptually weighted squared error is given by

$$d(\mathbf{x},\mathbf{y}) = (\mathbf{x}-\mathbf{y})^t\mathbf{W}(\mathbf{x}-\mathbf{y}), \tag{4.41}$$

where $\mathbf{W}$ is symmetric and positive definite matrix and the vectors $\mathbf{x}$ and $\mathbf{y}$ are treated as column vectors. Typically $\mathbf{W}$ is a diagonal matrix with diagonal values $w_{ii} > 0$; so we have

$$d(\mathbf{x},\mathbf{y}) = \sum_{i=1}^{k} w_{ii}(x_i - y_i)^2 \tag{4.42}$$

which is a simple but useful modification of the squared error distortion. The basic idea for this distortion measure was introduced by Gardner and Rao [76].

4.2 Phase Perception in Speech

In modern speech and audio coding technology, the application of perceptual characteristics of human auditory system is of great importance for efficient quantization of parameters. However, the focus on perception in coding technology has been confined only to the magnitude information of the signal, and little attention has been paid to phase information.

As noted in [77], in 1843 G. S. Ohm assumed that the phase of a waveform has no effect on how the ear perceives it. During the last century, this assumption has been disproved by a number of researchers [78, 79, 80]. For example, there exists a perceived difference between two different harmonic signals with same magnitude spectrum but with different phase spectrum.

It is known that the phase spectrum has a huge effect on the time-variation of sound pressure on the eardrum and thereby can change the perceived quality in modern speech coding systems [81]. Thus, it is desirable to quantify the perceptual redundancy of phase information for the efficient coding of speech or acoustic signals. As a result, several researchers have addressed this important issue [11, 82]. However, there is no comprehensive theory of phase perception yet, and the efficient compression and transmission of phase information is still an open problem.

In the early stage of our research work, we attempt to quantify the perceptual redundancy of phase information and thereby to encode only that part of phase information which is perceptually important. A novel idea in this field was published in [11], which proposed the *Perceptually Irrelevant Phase Elimination* (PIPE) criterion to determine the irrelevant phase information of acoustic signals. This method is based on the observation that the relative phase relationship within a critical band is perceptually important. PIPE criterion is particularly suitable for the harmonic signals. The critical phase frequency, for the harmonic signals, is defined and the phase information of the harmonic frequencies below the critical phase frequency is not perceptually important. Due to the harmonic structure of the slowly evolving waveform in the waveform interpolation coder, this method seems appropriate for our WI coder.

Before going to the details of PIPE criterion let us first develop the basics that are required to get better understanding of this criterion.

4.2.1 Critical Band Filtering

Auditory perception is based on the *critical band* analysis of the inner ear. The concept of the critical band, introduced by Fletcher [83], comes from the observation that the human auditory system shows a poorer frequency resolution at high frequencies than at low frequencies. This, together with other observations on masking of tones by noise [84, 85, 86], led to modelling the peripheral auditory analysis by a bank of overlapping bandpass linear

filters called *critical band filters.* This model postulates that sounds are pre-processed (on a nonlinear scale) by these filters, whose center frequency spacings and bandwidths increase with frequency as shown in Fig. 4.13. It is clear that each filter collects energy components from a wide range of frequencies, which are then lumped together for further processing.

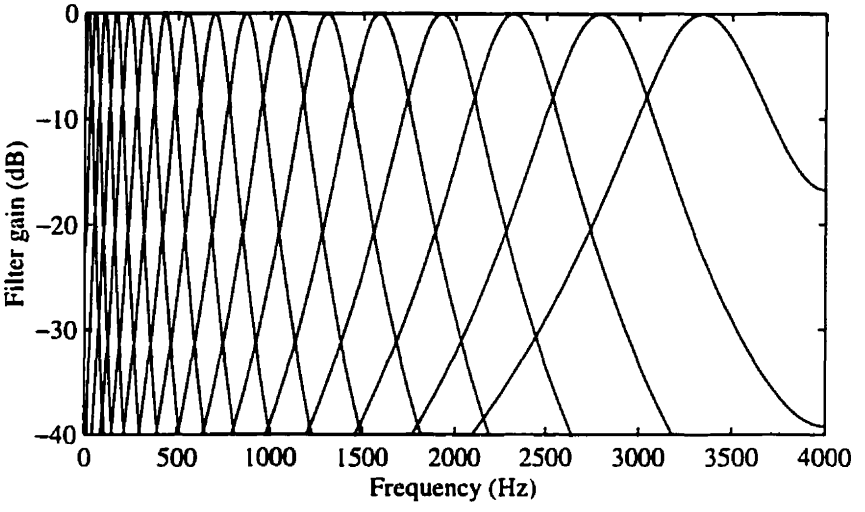


Fig. 4.13 Bank of critical-band (CB) filters. (From [87].)

The existence of the critical bands is related to the function of the basilar membrane. Each point on the basilar membrane is tuned (i.e., gives maximum response) to a frequency called the *characteristic frequency*, CF as shown in Fig. 4.14. However, the vibration of the membrane to a single frequency cannot be localized to an infinitely small area and nearby areas also show response to the same frequency, but with an amplitude that decreases with distance. So, each point on the BM can be considered as a bandpass filter with a certain center frequency (corresponding the CF) and a bandwidth (called the critical bandwidth). The bandwidth of these 'auditory filters' is not constant but increases with CF; their spacing corresponds to 1.5 mm [89] steps along the basilar membrane. Moore [27] defines a critical band as the Effective Rectangular Band (ERB), which is the bandwidth of an ideal bandpass filter centered at any frequency (the area under the squared-magnitude of the ideal filter equals that of the auditory filter centered at that frequency). According to Moore, each ERB covers 0.9 mm on the basilar membrane and it is given by [27]:

$$ERB = f/Q_{ear} + B_{\min}, \tag{4.43}$$

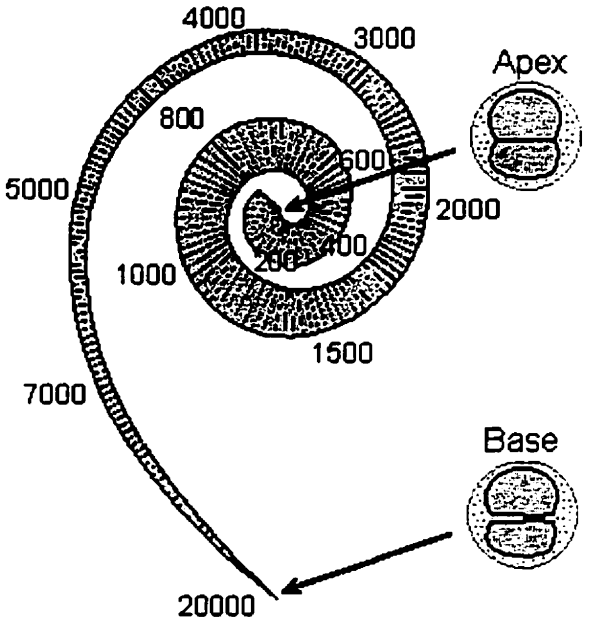


Fig. 4.14 Frequency-to-place transformation takes place in the cochlea, along the basilar membrane. (From [88].)

where f is frequency in Hz, Q_{ear} is the asymptotic filter quality at high frequencies and B_{min} is the minimum bandwidth for low frequency channels. Following Glasberg and Moore's suggestion, letting $Q_{ear} = 9.26449$ and $B_{min} = 24.7$ [90], the expression for ERB can be simplified as

$$ERB = 24.7(0.00437f + 1). \tag{4.44}$$

Observe that the frequency scale in Fig. 4.13 is linear, and at higher frequencies the critical-band filter shapes become wider (on a logarithmic scale, they have almost identical shapes at higher frequencies). In fact, the signal is processed in the inner ear on a nonlinear scale, called the *Bark scale*. (Bark is the unit of perceptual frequency and a critical band has a width of one Bark). It is conceptually convenient to think the Hertz-to-Bark transformation as the primary stage of critical-band filtering. Many analytical expressions have been proposed in the literature to relate the critical band number z (in Bark) to frequency f (in Hz). Schroeder *et al* in [89] propose the following formula

$$f = Y(z) = 650 \sinh(z/7). \tag{4.45}$$

Zwicker proposes the following [91]

$$z = Y^{-1}(f) = 13 \arctan(0.00076f) + 3.5 \arctan(f/7500)^2. \tag{4.46}$$

The mapping $Y(z)$ has been called the "critical-band density" [92]. According to Zwicker [91] the bandwidth of each critical band, as a function of center frequency, can be approximated by

$$\text{Critical Bandwidth} = 25 + 75(1 + 1.4(f/1000)^2)^{0.69}. \tag{4.47}$$

An example of critical bands covering a range of 3.7 kHz is listed in Table 4.4.

Table 4.4 List of critical bands covering a range of 3.7 kHz [91].

Band No.	Center Frequency (Hz)	Band (Hz)
1	50	0–100
2	150	100–200
3	250	200–300
4	350	300–400
5	450	400–510
6	570	510–630
7	700	630–770
8	840	770–920
9	1000	920–1080
10	1170	1080–1270
11	1370	1270–1480
12	1600	1480–1720
13	1850	1720–2000
14	2150	2000–2320
15	2500	2320–2700
16	2900	2700–3150
17	3400	3150–3700

4.2.2 Overview of the PIPE Criterion

One of the key concepts behind this technique is the impact of the phase spectrum on the shape of the envelope of the critical band signal which affects timbre perception. Timbre has been defined by the American Standards Association (1960) as the attribute of auditory sensation in terms of which listeners can judge sounds that have the same pitch but

dissimilar loudness. Timbre is roughly the property of any sound which classifies its source. The distinction can be coarse-grained, e.g, the difference between a tuba and a duck, or fine-grained, e.g., the distinction between my natural voice and my voice when I have a cold.

For better understanding of the derivation of the PIPE criterion let us define two terms: the phase change which modifies relative phase relationship within a critical band is termed *local phase change* (*within-channel phase change*), and the phase change which modifies the phase relationship between critical bands while maintaining phase relationship within a critical band, is termed *global phase change* (*between-channel phase change*). To derive the criterion, it is assumed (according to [93]) that the local phase change of the signal is perceptible but that the global phase change is negligible in terms of changes in timbre as it does not change the envelope of critical band signal [11].

Based on this philosophy, the criterion to quantify the perceptually irrelevant phase information is derived in [11] for Fourier signals as well as harmonic signals as follows. Consider the Fourier signal $x(t)$ of the form

$$x(t) = \sum_{k=1}^{K} A_k \cos(2\pi f_k t + \phi_k). \tag{4.48}$$

It is assumed that frequency f_k increases as the index k increases and $A_k \neq 0$. Now let us consider two adjacent critical bands $CB_L(f_k)$ and $CB_H(f_k)$ as shown in Fig. 4.15 and defined by

$$\begin{aligned} CB_L(f_k) &= \{f | f_{k,Low} \leq f \leq f_k\} \\ CB_H(f_k) &= \{f | f_k \leq f \leq f_{k,High}\}. \end{aligned} \tag{4.49}$$

If it is assumed that the magnitude response of the cochlear filter is rectangular in frequency domain, then critical bandwidth can be characterized by ERB given by Eq. (4.43). Consequently, two frequencies, $f_{k,Low}$ and $f_{k,High}$, can be calculated as

$$\begin{aligned} f_{k,Low} &= \frac{1}{2Q_{ear}+1}\left[f_k(2Q_{ear}-1) - 2Q_{ear}B_{\min}\right] \\ f_{k,High} &= \frac{1}{2Q_{ear}-1}\left[f_k(2Q_{ear}+1) + 2Q_{ear}B_{\min}\right]. \end{aligned} \tag{4.50}$$

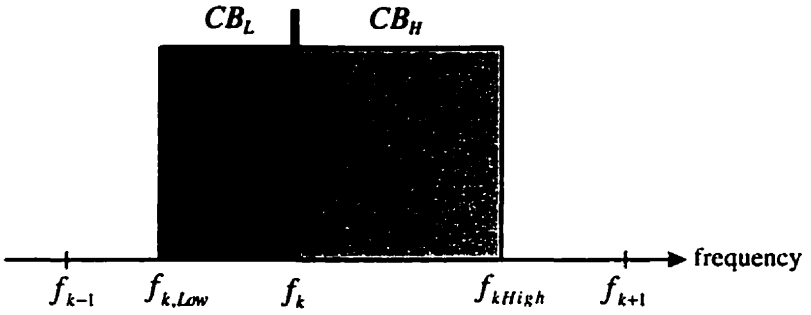


Fig. 4.15 Schematic diagram of two adjacent critical bands with f_k as the upper and lower bounds for Fourier signal.

Assuming $f_{k-1} < f_{k,Low} < f_k < f_{k,High} < f_{k+1}$, we see that whatever the value of ϕ_k it doesn't modify phase relationship within a critical band but changes only the between-channel phase relationship. Thus the phase corresponding to any frequency other than f_k corresponds to global phase change and would be perceptually irrelevant.

For the harmonic signal $f_k = kf_0$, where f_0 is the fundamental frequency. Suppose that there exists the j-th harmonic frequency, $f_j = jf_0$, for which the relative position of other harmonic frequencies are given in Fig. 4.16.

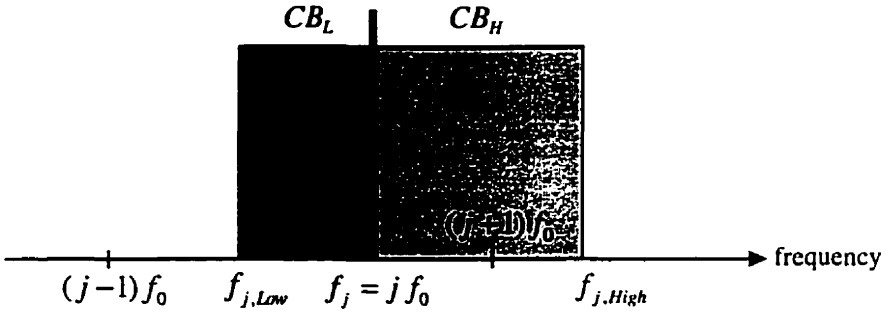


Fig. 4.16 Schematic diagram of two adjacent critical bands with f_k as the upper and lower bounds for harmonic signal.

For the harmonic component satisfying $(j-1)f_0 < f_{j,Low} < jf_0 < (j+1)f_0 < f_{j,High}$, using the same procedure followed for Fourier signal, it can be shown that only the phase components ϕ_k for $k \geq j$, are perceptually important for the signal consisting of fundamental frequency and its harmonics. By using Eq. (4.43) and Eq. (4.50) this phase index

can be calculated as follows [11]:

$$j = \left\lceil Q_{ear}\left(1 - \frac{B_{\min}}{f_0}\right) - 0.5 \right\rceil . \tag{4.51}$$

For harmonic signal having the fundamental frequency $f_0 = 100$ Hz, $j = 7$ which means that the phase corresponding to the frequencies $f_k < 700$ Hz is perceptually irrelevant and need not be transmitted. That's why f_j is called the *critical phase frequency*. It is clear that the critical phase frequency increases as the fundamental frequency increases. Therefore, there are relatively more perceptually important phase components for low-pitch signals (e.g., male speech) than for high-pitched (e.g., female speech) signals.

4.3 Summary

In this chapter we have discussed the concept of the two classes of product code vector quantization methods—the split VQ and the shape-gain VQ. In our WI coder we combine these two methods in a modified way to encode the SEW magnitude spectrum in an efficient manner. The concept of the critical phase frequency which is expected to quantify the perceptually irrelevant phase components of the SEW spectrum was also derived. The next chapter shows how we fit these two algorithms within the existing WI framework.

Chapter 5

Simulation and Results

This chapter presents the simulation and experimental results of the *split/shape-gain* vector quantization to quantize the amplitude of the Slowly Evolving Waveform (SEW) and that of the PIPE (Perceptually Irrelevant Phase Elimination) criterion to quantify perceptually irrelevant phase information of the SEW. The platform chosen to simulate and test the algorithms is the floating point C using Microsoft Visual C++ compiler.

5.1 Framework for SEW Magnitude Quantization

In this thesis we present a novel technique to quantize the SEW magnitude information in a more efficient manner. Our main concern is the improvement of processors **344–346 & 444–446** in Fig. 3.8. The improvement is realized by introducing a hybrid product code vector quantization method. Denoted as *split/shape-gain VQ*, it combines the split VQ and the shape-gain VQ. The algorithm for this new method is presented below.

Step-1: Divide the SEW magnitude vector $\mathbf{X} \in \mathcal{R}^n$, having a bandwidth of 4 kHz, into three subvectors $\mathbf{X}_1 \in \mathcal{R}^{n_1}$, $\mathbf{X}_2 \in \mathcal{R}^{n_2}$, $\mathbf{X}_3 \in \mathcal{R}^{n_3}$ representing 0–1000 Hz, 1000–2000 Hz and 2000–4000 Hz respectively as shown in Fig. 5.1. Here $n = n_1 + n_2 + n_3$.

Step-2: Compute the gain of each subvector.

$$\text{gain}, \quad g_j = \|\mathbf{X}_j\| = \sqrt{\sum_{i=1}^{n_j} X_{j,i}}, \qquad \text{for } j = 1, 2, 3 \tag{5.1}$$

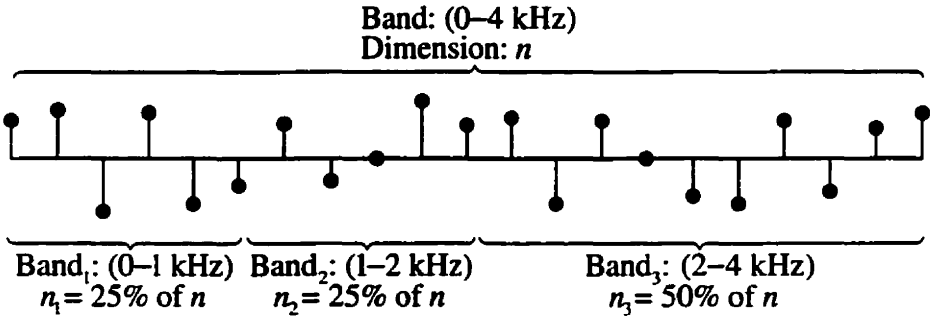


Fig. 5.1 Splitting of a Slowly Evolving Waveform (SEW) into three subbands.

where we have used the fact that $\mathbf{X}_j = (X_{j,1}, X_{j,2}, \ldots, X_{j,n_j})$ for $j = 1, 2, 3$.

Step-3: Quantize gain vector (g_1, g_2, g_3) as $(\hat{g}_{p,1}, \hat{g}_{p,2}, \hat{g}_{p,3})$, $1 \leq p \leq N_g$ by selecting the nearest neighbor from the gain codebook $\mathcal{C}_g = (\hat{g}_1, \hat{g}_2, ..., \hat{g}_{N_g})^T$, where N_g is the size of the gain codebook and the i-th codeword is given by $\hat{g}_i = (\hat{g}_{i,1}, \hat{g}_{i,2}, \hat{g}_{i,3})$ for $1 \leq i \leq N_g$. The Gain codebook is designed using GLA with the mean squared error as the distortion measure. Since as is well known, the logarithm of the signal power is perceptually more relevant than the signal power itself, the gain quantization is performed in the logarithmic domain.

Step-4: Normalize each subvector by its corresponding quantized gain value to produce the shape vector.

$$\text{shape vector,} \quad \mathbf{S}_j = \frac{\mathbf{X}_j}{\hat{g}_{p,j}} \qquad \text{for } j = 1, 2, 3 \text{ and } 1 \leq p \leq N_g. \tag{5.2}$$

Step-5: Quantize the shape vector $\mathbf{S}_j$ as $\hat{\mathbf{S}}_q$, $1 \leq q \leq N_s$ by choosing the nearest neighbor from the shape codebook $\mathrm{C_s}= (\hat{\mathbf{S}}_1, \hat{\mathbf{S}}_2, ..., \hat{\mathbf{S}}_{N_s})^T$ for $1 \leq i \leq N_s$, where N_s is the size of the shape codebook. The shape codebook is designed using GLA along with a perceptually based bit allocation strategy and perceptually-weighted mean squared error as the distortion measure. Perceptual weighting factor is calculated in the way followed in the existing WI coder as is presented in Section 3.2.4.

Step-6: At the receiver, the discontinuity of the gain contour at the boundary of subframes introduces buzziness in the reconstructed speech. This problem is tackled by

smoothing the gain contour using a piecewise monotonic cubic interpolant. To facilitate the use of the monotonic interpolant, it is assumed that the gain of the leftmost (rightmost) harmonic in the first (third) subband is the average gain of that subband; for the middle subband it is at the middle of the corresponding subband.

The block-diagram representation of the above-stated *split/shape-gain VQ* design steps is presented in Fig. 5.2.

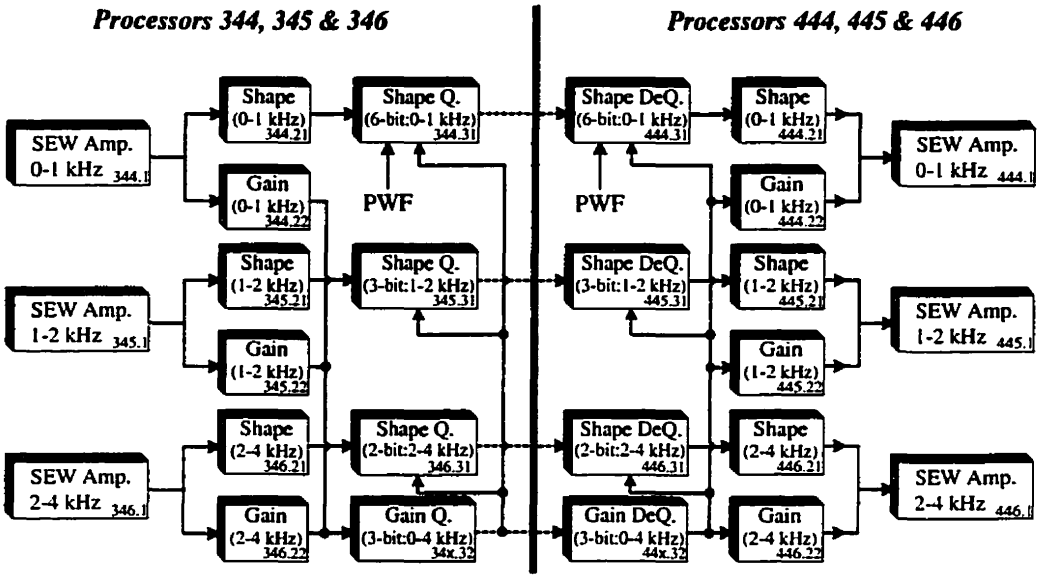


Fig. 5.2 The block-diagram of the proposed split/shape-gain VQ for efficient SEW (amplitude) coding.

Setup for Quantizer Design

The training sequence for designing the SEW gain and shape codebooks requires large database. The speech database used in our research consists of 15 min of speech recorded from 23 different male and female speakers uttering in total of 325 sentences. The 325 speech files are concatenated in such a way that male and female speakers appear alternately. This large speech file, when transmitted through the analysis part of our WI coder, gives 81,600 SEW shape vectors (for each of the three bands as stated above) and the same number of gain vectors which are used as the training vectors to design the respective codebooks. This length of the training sequence is assumed to be sufficiently large so that all the statistical

properties of the source are captured by the training sequence. For testing the performance of the new codebooks in coding the SEW amplitude spectrum we use another 10 sentences from 5 different males and 5 different females (different from those used for training).

The dimension of the SEW amplitude spectrum depends on the interpolated pitch period of the corresponding subframe. Consequently, the spectrum has a variable dimension. To tackle this variable dimension issue in our shape and gain codebooks design process we incorporate the following dimension conversion technique.

- In our implementation, the pitch is allowed to vary from 20 to 120 resulting in 10 to 60 harmonics in the SEW magnitude spectrum (the DC component is excluded). Thus, each of the two subbands may have up to 15 harmonics while the baseband has up to 30 harmonics. Prior to codebook training, each SEW shape vector in the training set is first bandlimited interpolated to a fixed dimension. A natural choice for the dimension of the vectors of the first two subbands is 15 each and that of the baseband is 30. Here it is assumed that a variable-dimension vector is generated as a result of uniform sampling of another vector with a fixed and large dimension. Instead of the gain itself, the use of "gain per dimension per subband" as the element of gain vector in the training set circumvents the variable dimension issue in gain codebook design. Now, the conventional GLA technique is applied separately on each of the gain and the shape training sets.

- When encoding a shape spectrum in processors **344.21, 345.21, and 346.21**, the respective shape-codebook is first subsampled to the length of the given spectrum. Then a nearest neighbor search is carried out using the perceptually-weighted mean squared error criterion.

With the constraint that the overall bit consumption for each SEW must be 14 as in the existing 4.25 kbps WI coder, we follow a perceptually based bit allocation strategy. With the thought that human ear has better resolution capability at lower frequencies we use maximum bits (6 bits) to encode first subband (0–1 kHz). The second subband is encoded with 3 bits. At this stage we have 5 bits to encode both the gain vector of the three bands and the shape vector of the baseband. The performance of the proposed *split/shape-gain VQ* method is evaluated for the following two setups.

Setup–1: A 3-bit shape codebook for baseband and a 2-bit gain codebook.

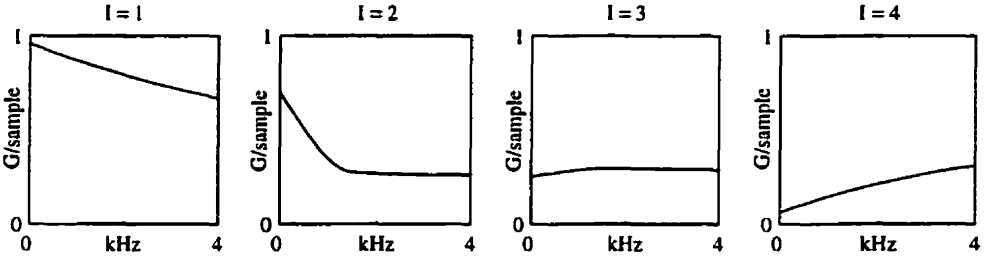
Setup–2: A 2-bit shape codebook for baseband and a 3-bit gain codebook.

5.1.1 Simulation Results

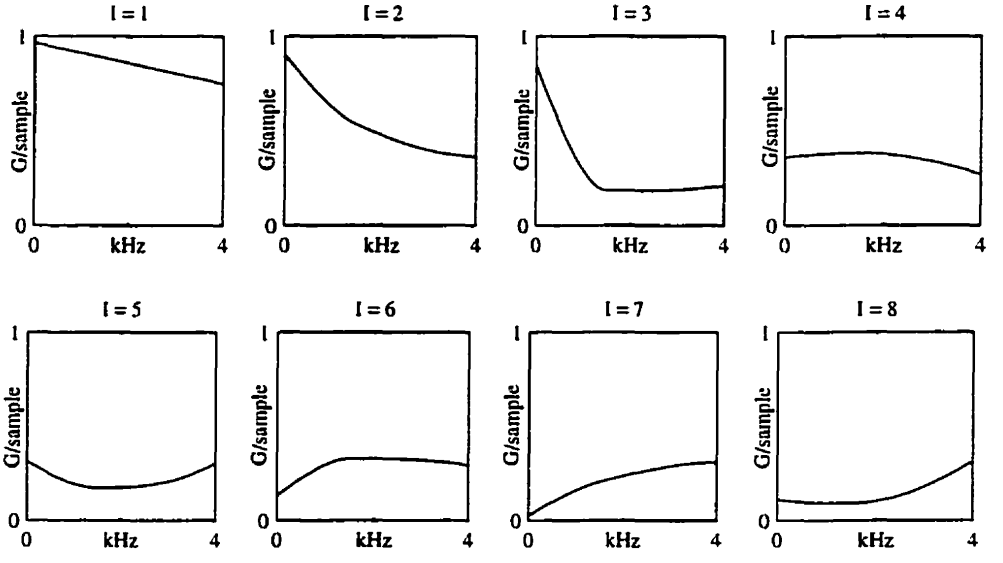
This section presents the results of tests carried out to measure the performance of the proposed algorithm for the two setups as stated above.

A relative threshold value of 0.00001 or a maximum number of iterations of 1000 is used as the stopping criterion for the GLA. However, in our simulations, the latter situation has never been attained signifying quick convergence of the GLA. The GLA exploits the perceptually weighted Euclidean distance measure discussed in Section 4.1.4, where the perceptual weighting factor is calculated using the procedure described in Section 3.2.4. The four (eight) possible shapes of the gain contours of the SEW magnitude spectrum for setup–1 (setup–2) are shown in Fig. 5.3(a) (Fig. 5.3(b)). The contours play a key role in producing the high quality of the synthesized speech of WI coders. In Fig. 5.3(a) and Fig. 5.3(b), lower indices represent voiced signals while higher indices represent unvoiced signals. Thus for index, I = 1, the normalized spectrum is filled with the high-energy slowly evolving waveform. As the indices get higher, the amount of SEW decreases to accommodate more REW. Further, it may also be seen that these contours suggest that high frequency regions of the SEW spectrum are more random than low frequency regions (hence the higher amount of REW at the higher frequencies). The voiced/unvoiced decision in the WI paradigm exploits the fact that the energy of the SEW dominates in the voiced region whereas that of the REW dominates in the unvoiced region. Therefore, the degree of voicing is roughly proportional to the SEW/CW energy ratio (or inversely proportional to the REW/CW energy ratio). It is clear from Fig. 5.4 that a threshold value of 0.55 for SEW/CW energy ratio performs well for the voiced/unvoiced decision.

The human auditory system is the ultimate evaluator of the quality and performance of a speech coder in preserving intelligibility and naturalness. We conducted informal subjective tests to compare the two setups between themselves and with the existing one. The evaluation of the reconstructed speech was carried out on 10 English sentences spoken by 5 male speakers and 5 female speakers, and judged by 5 listeners. The test results show that increasing the size of the SEW gain codebook from 2-bit to 3-bit gives marginal improvement in output speech quality. During informal listening tests the speech is reported as sounding smoother and natural. However, both the setups are reported to have im-



(a) Four gain contours in 2-bit gain codebook.



(b) Eight gain contours in 3-bit gain codebook.

Fig. 5.3 Shapes of the gain contours obtained by using monotonic cubic interpolation method on gain (G) per sample per band for 2 & 3-bit gain codebooks.

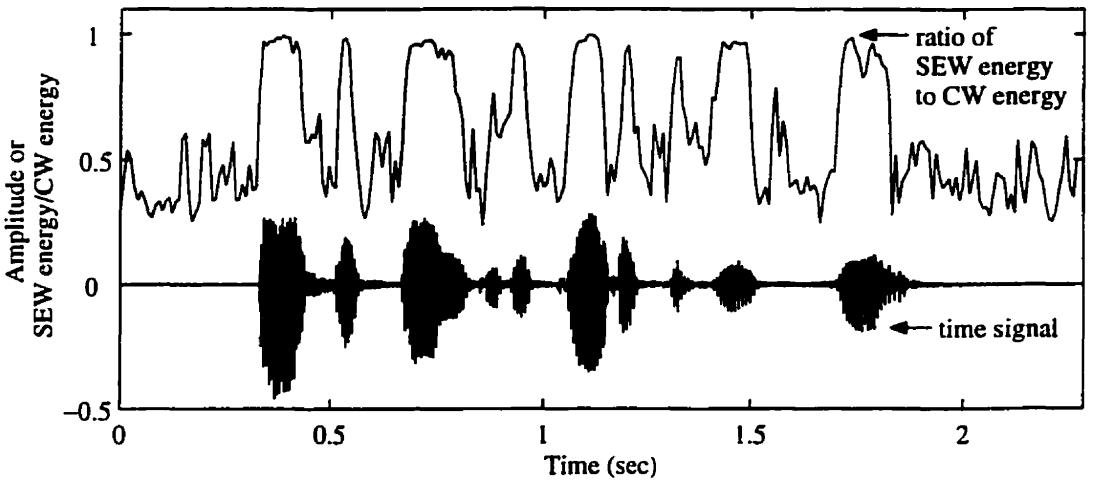


Fig. 5.4 Voiced/unvoiced detection in WI paradigm. A threshold value of 0.55 for SEW/CW energy ratio performs well for the voiced/unvoiced decision.

proved output speech quality compared to that of the existing WI model. The candidate method improves the reconstructed speech quality as is evident from Fig. 5.5. It shows that split/shape-gain VQ method produces synthesized speech that most closely resembles the original input speech, most notably in transitions. This is due to the better matching of the SEW from the combination of the shape and the gain codebooks.

5.2 Framework to Verify the PIPE Criterion

To investigate the validity of the PIPE (Perceptually Irrelevant Phase Elimination) criterion in the waveform interpolation speech coding system, we have determined the critical phase frequency, f_j, using Eq. (4.51) for each slowly evolving waveform (SEW), which is characterized by estimated fundamental frequency corresponding to that subframe. Only the phase components ϕ_k, for $k \geq j$, which are perceptually important according to the PIPE criterion are transmitted. In the receiver, the phase information ϕ_k, for $k < j$, is estimated using quadratic interpolation.

5.2.1 Simulation Results

The evaluation of the reconstructed speech incorporating the unquantized version of perceptual phase processing for SEW and full phase (0 to 4 kHz) processing for REW was

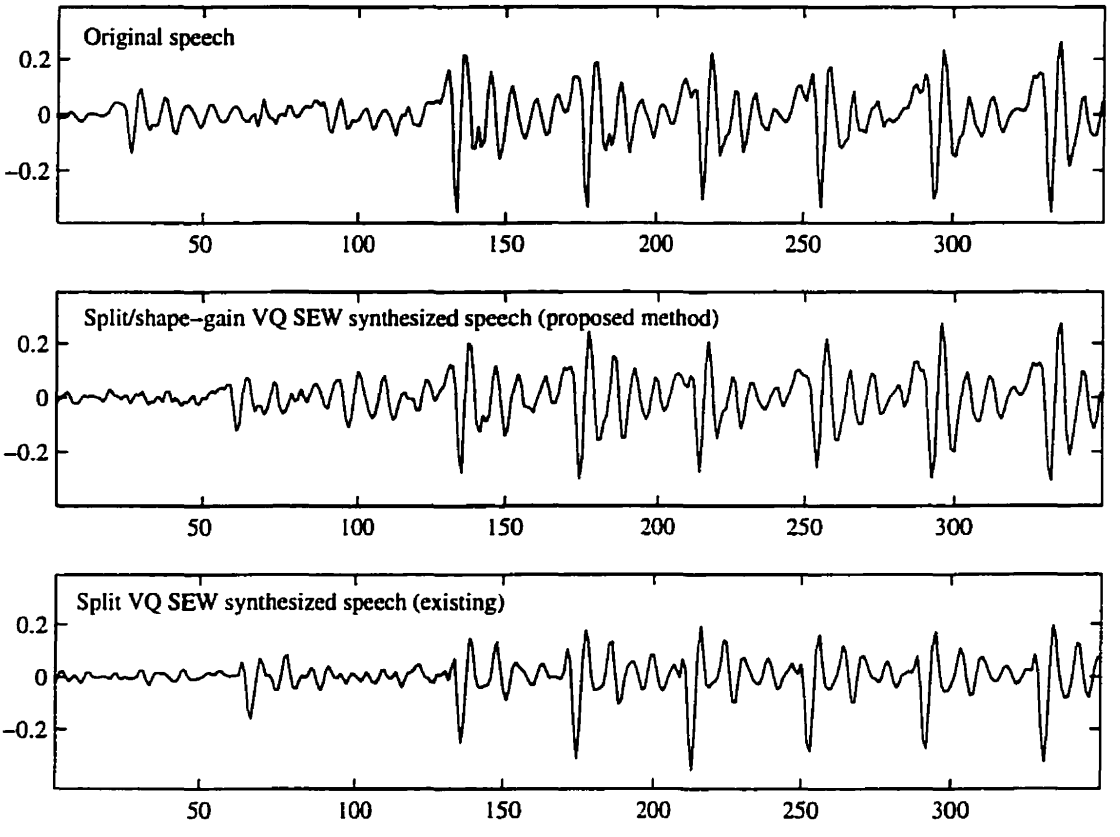


Fig. 5.5 Reconstructed speech comparison. The *split/shape-gain* VQ method improves the reconstructed speech quality, most notably in transitions.

carried out by informal subjective test on 10 English sentences from 5 male speakers and 5 female speakers, judged by 5 listeners. Unlike Kim's [11] result, our result shows that the inclusion of partial phase information determined by the PIPE criterion does not seem to produce same quality speech compared to the output speech with full phase transmitted. It is also observed that the output speech with full SEW and REW phase transmitted offers very little improvement over that with no SEW and REW phase transmitted. From these observations and informal subjective test results, we can infer that:

- The PIPE criterion which claims that "global phase change is perceptually irrelevant" is not valid. This is invalid not only in the WI paradigm but also for pure harmonic signal. We verified this by carrying out the same test using a pure harmonic signal as Kim [11] did. In this setup, we perturbed the phase of the first six harmonics (one harmonic per each critical band) of a pure harmonic signal with fundamental frequency 200 Hz and bandwidth 3800 Hz; the reconstructed signal was reported to be significantly different from the original one.

- In a parametric coder (e.g., WI coder), the phase information is of secondary perceptual significance, particularly for low bit-rate speech coding and hence need not be transmitted. The most recent work on the SEW quantization also supports our claim [94].

5.2.2 Importance of Phase and the PIPE Criterion

The basic idea of the PIPE criterion stems from a revised premise of the Lipshitz paper [77] presented by Hartmann [95]. Here, the author states that: "In general, the relative phase between two signal components should be irrelevant if the two components are separated by more than a critical bandwidth. Phase should not matter then because there is no neural element (of the cochlea) that sees both components." However, this revised argument is also inconsistent with results discussed below regarding tones an octave apart[1] - which is more than a critical bandwidth.

Hartmann notes that if non-linearities are present, the relative phase between two frequencies producing intermodulation distortion can drastically affect the amplitude of the distortion. In fact, as he shows, in the case of intermodulation between a fundamental tone

[1]See the notes on "Audibility of Phase Distortion" by A. Ludwig in [96].

and its second harmonic, it is possible for the fundamental tone to completely disappear for certain (improbable) combinations of amplitude and phase! Therefore, the level of distortion produced by a loudspeaker can depend on the phase response of a crossover feeding it. But the distortion could be either higher or lower, so this argument says that a flat phase response is different, not better, than a non-flat response. However, it doesn't address the issue of audibility of the differences.

Regarding the cochlea non-linearities, Hartmann states that if two tones are in different critical bands, intermodulation products are not expected, and thus would not introduce phase sensitivity. However, harmonic distortion products due to ear nonlinearities do introduce phase sensitivity. It is well known that a loud tone can make an otherwise audible weaker tone completely inaudible (masking). In a very interesting paper, Nelson and Bilger [97] show that the masking level of a tone for its second harmonic depends on the relative phase of the tones. The difference is as large as 30 dB. Apparently, this is a result of a second harmonic produced by nonlinearity in the ear itself, adding constructively or destructively with the externally produced second harmonic. As noted in their paper, this effect varies quite a bit among individuals. Some people might not detect much difference. But for some other people this effect could change perception of harmonic distortion produced by a sound system, again for better or worse. This effect can also alter perception of harmonics naturally produced by musical instruments, so here we have a valid argument that fidelity can be audibly degraded by a realistic phase change, but not in the way as is assumed in developing the PIPE criterion.

Chapter 6

Conclusions and Future Work

To meet state-of-the-art requirements for robustness and high quality reconstructed speech at low bit-rates, it is necessary to focus the limited coding resources on perceptually relevant features of the speech signal. This thesis has described a coding method aimed at satisfying these requirements. We have particularly dealt with the slowly evolving part (SEW) that characterizes the voiced part of the speech signal. Both the magnitude and the phase information have been treated separately.

6.1 Conclusion: SEW Magnitude Quantization

The proposed split/shape-gain VQ method facilitates efficient parameterization of SEW magnitude by preserving the critical attributes of the voiced part of speech signal and thereby improves the quality of the reconstructed speech compared to that of the existing (slightly rough sounding) WI coder at 4.25 kbps. It provides a speech quality which converges to that of the 8 kbps G.729 with increasing bit allocation for SEW gain codebook and therefore this scheme achieves very close to toll quality. However, the upper bound on the size of the SEW gain codebook can be set to 3 bits (provided the bit-budget for both the SEW gain codebook and the baseband (2–4 kHz) SEW shape codebook is 5 bits) as beyond that the overall quality deteriorates due to the low availability of bits to represent the SEW spectral shape vector properly within the 4.25 kbps framework. The bit-allocation scheme is summarized in Table 6.1. Without any speed optimization, the current implementation runs in approximately real time on a Pentium-II 350 MHz processor.

Table 6.1 Bit allocation for the 4.25 kbps WI coder.

WI parameters	Bits/update	Update rate	Bits/frame	Bits/sec.
Pitch	7	50 Hz	7	350
Power	4	100 Hz	8	400
LSFs	30	50 Hz	30	1500
SEW (shape)	(6+3+2)	100 Hz	22	1100
SEW (gain)	3	100 Hz	6	300
SEW (phase)	0	100 Hz	0	0
REW (amplitude)	3	200 Hz	12	600
REW (phase)	0	200 Hz	0	0
		Overall bit-rate	85	**4250**

6.2 Conclusion: SEW Phase Quantization

At the outset of our research work we attempt to quantify the perceptually irrelevant phase components using the Perceptually Irrelevant Phase Elimination (PIPE) criterion. However, we have shown that the PIPE criterion fails to predict the importance of phase. We have also shown that the inclusion of unquantized phase components for the entire speech band does not improve the quality of the output speech when compared to speech with no SEW-phase. Thus, we infer that no phase components and hence no phase quantizer are necessary for the SEW in the WI model, particularly in low bit-rates.

6.3 Suggestions for Further Investigation

In this section, we enumerate issues which show promise to improve the quality of the WI reconstructed speech.

- It is reported in [42] that LSFs can be quantized very efficiently using split-VQ at 24 bits/frame. Since our WI coder encodes LSFs at 30 bits/frame, we can save 6 bits/frame which can be spent to encode the SEW shape vectors and the gain vectors using larger and hence more accurate codebooks.

- We should pay attention to SEW/CW (or REW/CW) energy ratio. This information may be transmitted as a side information.

- We have, in fact, employed open loop quantization schemes for SEW and REW

magnitudes. There are many other closed-loop quantization schemes [35, 98], which successfully incorporate the analysis-by-synthesis technique into the WI coding.

- There is no comprehensive theory of phase perception yet, and the efficient compression and transmission of phase information is still an open problem.

References

[1] P. Rubin and E. V. Bateson, *Measuring and modeling speech production.* Yale University, 1998. *http://www.haskins.yale.edu/haskins/HEADS/MMSP/intro.html.*

[2] D. O'Shaughnessy, *Speech Communications: Human and Machine.* New York: IEEE Press, second ed., 2000.

[3] *Compton's Encyclopedia.* Proteus Enterprises Inc, 1999.

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

[5] A. Gersho, "Advances in Speech and Audio Compression," *Proceedings of the IEEE*, vol. 82, pp. 900–918, June 1994.

[6] P. Mermelestein, "G.722, a New CCITT Coding Standard for Digital Transmission of Wideband Audio Signals," *IEEE Commun. Mag.*, vol. 26, pp. 8–15, Jan. 1988.

[7] A. S. Spanias, "Speech Coding: A Tutorial Review," *Proceedings of the IEEE*, vol. 82, pp. 1541–1582, Oct. 1994.

[8] K. Jarvinen, J. Vainio, P. Kapanen, T. Honkanen, P. Haavisto, R. Salami, C. Laflamme, and J. P. Adoul, "GSM Enhanced Full Rate Speech Codec," *Proc. IEEE Int. Conf. on Acoustics, Speech, Signal Processing* (Munich), vol. 2, pp. 771–774, 1997.

[9] T. Honkanen, J. Vainio, K. Jarvinen, P. Haavisto, R. Salami, and C. L. J. P. Adoul, "Enhanced Full Rate Speech Codec for IS-136 Digital Cellular System," *Proc. IEEE Int. Conf. on Acoustics, Speech, Signal Processing* (Munich), vol. 2, pp. 731–734, 1997.

[10] A. DeJaco, W. Gardner, P. Jacobs, and C. Lee, "QCELP:The North American CDMA Digital Cellular Variable Rate Speech Coding Standard," *Proc. IEEE Workshop on Speech Coding for Telecommunications* (Ste. Adele), pp. 5–6, 1993.

[11] D. S. Kim, "Perceptual phase redundancy in speech," *Proc. IEEE Int. Conf. on Acoustics, Speech, Signal Processing* (Istanbul, Turkey), vol. 3, pp. 1383–1386, May 2000.

[12] W. B. Kleijn, "Continuous representations in linear predictive coding," *Proc. IEEE Int. Conf. on Acoustics, Speech, Signal Processing* (Toronto), vol. 1, pp. 201–204, May 1991.

[13] E. L. T. Choy, "Waveform Interpolation Speech Coder at 4 kb/s," Master's thesis, McGill University, Montreal, Canada, Aug. 1998.

[14] G. H. Golub and C. F. Van Loan, *Matrix Computations.* Baltimore, Maryland: The John Hopkins University Press, third ed., 1996.

[15] S. Haykin, *Adaptive Filter Theory.* Upper Saddle River, New Jersey: Prentice Hall, third ed., 1996.

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

[17] R. Hagen, E.Paksoy, and A. Gersho, "Voicing-specific lpc quantization for variable-rate speech coding," *IEEE Trans. Speech and Audio Processing*, vol. 7, pp. 485–494, Sept. 1999.

[18] M. S. Lee and H. K. K. and; Hwang Soo Lee, "Lpc analysis incorporating spectral interpolation for speech coding," *Electronics Letters*, vol. 35, pp. 200–201, Feb. 1999.

[19] F. Itakura, "Line spectrum representation of linear predictive coefficients of speech signals," *Journal Acoustical Society America*, vol. 57, p. S35, Apr. 1975. abstract.

[20] F. K. Soong and B.-H. Juang, "Line Spectrum Pair (LSP) and speech data compression," *Proc. IEEE Int. Conf. on Acoustics, Speech, Signal Processing* (San Diego, California), pp. 1.10.1–1.10.4, Mar. 1984.

[21] G. S. Kang and L. S. Fransen, "Application of line spectrum pairs to low bit rate speech encoders," *Proc. IEEE Int. Conf. on Acoustics, Speech, Signal Processing* (Tampa, Florida), pp. 7.3.1–7.3.4, Apr. 1985.

[22] G. S. Kang and L. J. Fransen, *Low-bit rate speech encoders based on line spectrum frequencies (LSFs).* Naval Research Laboratory Report 8857, Nov. 1984.

[23] P. Kabal and R. P. Ramachandran, "The computation of line spectral frequencies using chebyshev polynomials," *IEEE Trans. Acoustics, Speech, Signal Processing*, vol. ASSP-34, pp. 1419–1426, Dec. 1986.

[24] A. H. *et al*, "Articulation testing methods," *J. Acoust. Soc. Am.*, vol. 37, pp. 158–166, 1966.

[25] R. Kubichek, "Standards and Technology Issues in Objective Voice Quality Assessment.," *Digital Signal Processing: A Review Journal.*, vol. DSP, pp. 38–44, Apr. 1991.

[26] S. Wang, A. Sekey, and A. Gersho, "An objective measure for predicting subjective quality of speech coders," *IEEE J. Selected Areas in Comm.*, vol. 10, pp. 819–829, June 1992.

[27] B. C. J. Moore, *An Introduction to the Psychology of Hearing.* Academic Press, fourth ed., 1997.

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

[29] S. R. Quackenbush, T. P. Bamwell, and M. A. Clements, *Objective measures for speech quality.* Englewood Cliffs, New Jersey: Prentice-Hall, 1988.

[30] I. Cotanis, "Speech quality evaluation for mobile networks," *Proc. IEEE Int. Conf. On Communications* (Glasgow, UK), vol. 3, pp. 1530–1534, May 2000.

[31] J. Beerends and J. Stemmerdink, "A perceptual speech quality measure based on a psychoacoustic sound representation," *J. Audio Eng. Soc.*, vol. 40, pp. 963–978, 1992.

[32] ITU-T, *Objective Quality Measurement of the telephone band (300 Hz - 3400 Hz) speech codecs*, Feb. 1998. ITU-T Recommendation P.861.

[33] W. B. Kleijn, "Encoding speech using prototype waveforms," *IEEE Trans. Speech and Audio Processing*, vol. 1, pp. 386–399, Oct. 1993.

[34] W. B. Kleijn and K. K. Paliwal, eds., *Speech Coding and Synthesis.* Amsterdam: Elsevier, 1995. Ch. 5: Waveform Interpolation for Coding and Synthesis.

[35] I. Burnett, "The waveform interpolation paradigm. Foundation of a class of speech coders," *TENCON '97. IEEE Region 10 Annual Conference. Speech and Image Technologies for Computing and Telecommunications., Proceedings of IEEE*, vol. 1, p. 1, 1997.

[36] Telecommunications Industry Association, TIA/EIA/PN-3292. *EIA/TIA Interim Standard, Enhanced Variable Rate Codec (EVRC)*, Mar. 1996. ITU-T Recommendation P.861.

[37] W. B. Kleijn and J. Haagen, "A speech coder based on decomposition of characteristic waveforms ," *Proc. IEEE Int. Conf. on Acoustics, Speech, Signal Processing*, vol. 1, pp. 508–511, 1995.

[38] D. J. Hiotakakos and C. S. Xydeas, "Low bit rate coding using an interpolated zinc excitation model," *ICCS*, vol. 3, pp. 865–869, Oct. 1994.

[39] Y. Hiwasaki and K. Mano, "A new 2-kbit/s speech coder based on normalized pitch waveform," *Proc. IEEE Int. Conf. on Acoustics, Speech, Signal Processing*, vol. 2, pp. 1583–1586, 1997.

[40] M. Leong, "Representing voiced speech using prototype waveform interpolation for low-rate speech coding," Master's thesis, McGill University, Montreal, Canada, Nov. 1992.

[41] R. Drullman, J. Festen, and R. Plomp, "Effect of temporal envelope smearing on speech reception," *J. Acoust. Soc. Am.*, vol. 95, no. 2, pp. 1053–1064, 1994.

[42] K. K. Paliwal and B. S. Atal, "Efficient vector quantization of LPC parameters at 24 bits/frame," *IEEE Trans. Speech and Audio Processing*, vol. 11, pp. 3–14, Jan. 1993.

[43] G. Kubin, B. S. Atal, and W. B. Kleijn, "Performance of noise excitation for unvoiced speech," *Proc. IEEE Workshop on Speech Coding for Telecommunications* (Sainte-Adele, Quebec), pp. 35–6, 1993.

[44] O. Gottesman and A. Gersho, "Enhanced waveform interpolative coding at 4 kbps," *Proc. IEEE Workshop on Speech Coding*, pp. 90–92, 1999.

[45] C. E. Shannon, "A mathematical theory of communication," *Bell Syst. Tech. J.*, vol. 27, pp. 379–423 and 623–656, 1948.

[46] C. E. Shannon, "Coding theorems for a discrete source with a fidelity criterion," *IRE National Convention record, part 4*, pp. 142–163, 1959.

[47] R. E. Blahut, "Computation of Channel Capacity and Rate-Distortion Function," *IEEE Trans. Inform. Theory*, vol. 18, pp. 460–473, July 1972.

[48] N. S. Jayant and P. Noll, *Digital Coding of Waveforms*. Prentice Hall, 1984.

[49] J. MacQueen, "Some methods for classification and analysis of multivariate observations," *Proc. of the Fifth Berkeley Symposium on Math. Stat. and Prob.*, vol. 1, pp. 281–296, 1967.

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

[51] Y. Linde, A. Buzo, and R. Gray, "An Algorithm for Vector Quantizer Design," *IEEE Trans. Communications*, vol. 28, pp. 84–95, Jan. 1980.

[52] N. B. Karayiannis and J. C. Bezdek, "An integrated approach to fuzzy learning vector quantization and fuzzy c-means clustering," *IEEE Trans. Fuzzy Systems*, vol. 5, pp. 622–628, Nov. 1997.

[53] W. H. Equitz, "A new vector quantization clustering algorithm," *IEEE Trans. Acoustics, Speech, Signal Processing*, vol. 37, pp. 1568–1575, Oct. 1989.

[54] N. B. Karayiannis and P. I. Pai, ""Fuzzy vector quantization algorithms and their applications in image processing," *IEEE Trans. Image Processing*, vol. 4, pp. 1193–1201, Sept. 1995.

[55] K. Zeger and A. Gersho, "A stochastic relaxation algorithm for improved vector quantizer design," *Electronics Letters*, vol. 25, pp. 896–898, July 1989.

[56] J. K. Flanagan, D. R. Morrell, R. L. Frost, C. J. Read, and B. E. Nelson, "Vector quantization codebook generation using simulated annealing," *Proc. IEEE Int. Conf. on Acoustics, Speech, Signal Processing* (Gasgow, Scotland), vol. 3, pp. 1759–1762, May 1989.

[57] E. Yair, K. Zeger, and A. Gersho, "Conjugate gradient methods for designing vector quantizers," *Proc. IEEE Int. Conf. on Acoustics, Speech, Signal Processing*, vol. 1, pp. 245–248, 1990.

[58] K. Rose, E. Gurewwitz, and G. C. Fox, "A deterministic annealing approach to clustering," *Pattern Recognition Letters*, vol. 11, pp. 589–594, 1990.

[59] D. E. Rumelhart and D. Zesper, "Feature discovery by competitive learning," *Cognitive Sci.*, vol. 9, pp. 75–112, 1985.

[60] E. Yair, K. Zeger, and A. Gersho, "Competitive learning and soft competition for vector quantizer design," *IEEE Trans. Signal Processing*, vol. 40, pp. 294–309, Feb. 1992.

[61] D. Desieno, "Adding a conscience to competitive learning," *Proc. Int. Joint Conf. Neural Networks* (San Diego), vol. 1, pp. 117–124, June 1988.

[62] S. C. Ahalt, A. K. Krishnamurthy, P. Chen, and D. E. Milton, "Vector quantization using frequency-sensitive competitive-learning neural networks," *IEEE Int. Conf. on Systems Eng.*, pp. 131–134, 1989.

[63] N. M. Nasrabadi and Y. Feng, "Vector quantization of images based upon the Kohonen Self-Organization Feature Maps," *Proc. IEEE Int. Conf. Neural Networks*, vol. 1, pp. 101–108, 1988.

[64] J. C. Bezdek, E. C.-K. Tsao, and N. R. Pal, "Fuzzy Kohonen clustering networks," *Proc. Int. Joint Conf. Neural Networks*, pp. 1035–1043, 1992.

[65] T. M. Martinetz, S. G. Berkovich, and K. J. Schulten, ""neural-gas" network for vector quantization and its application to time-series prediction," *IEEE Trans. Neural Networks*, vol. 4, pp. 558–596, July 1993.

[66] J. Makhoul, S. Roucos, and H. Gish, "Vector Quantization in Speech Coding," *Proceedings of the IEEE*, vol. 73, pp. 34–69, Nov. 1985.

[67] T. D. Lookabaugh and R. M. Gray, "High-resolution quantization theory and the vector quantizer advantage," *IEEE Trans. Inform. Theory*, vol. 35, no. 5, pp. 1020–1033, 1989.

[68] A. Gersho, "Asymptotically optimal block quantization," *IEEE Trans. Inform. Theory*, vol. IT-25, no. 4, pp. 373–380, 1979.

[69] E. S. Barnes and N. J. A. Sloane, "The optimal lattice quantizer in three dimensions," *SIAM J. Algebraic Discrete Methods*, vol. 4, pp. 30–41, Mar. 1983.

[70] J. H. Conway and N. J. A. Sloane, "Voronoi regions of lattices, second moments of polytopes, and quantization," *IEEE Trans. Inform. Theory*, vol. IT-28, pp. 211–226, Mar. 1982.

[71] M. J. Sabin and R. M. Gray, "Product code vector quantizers for waveform and voice coding," *IEEE Trans. Acoustics, Speech, Signal Processing*, vol. ASSP-32, pp. 474–488, June 1984.

[72] W. Y. Chan, "The design of generalized product-code vector quantizers," *Proc. IEEE Int. Conf. on Acoustics, Speech, Signal Processing*, vol. 3, pp. 389–392, 1992.

[73] A. H. Gray, Jr. and J. D. Markel, "Distance measures for speech processing," *IEEE Trans. Acoustics, Speech, Signal Processing*, vol. ASSP-24, pp. 380–391, Oct. 1976.

[74] N. Batri, "Robust Spectral Parameter Coding in Speech Processing," Master's thesis, McGill University, Montreal, Canada, May 1998.

[75] R. M. Gray and D. L. Neuhoff, "Quantization," *IEEE Trans. Inform. Theory*, vol. 44, pp. 2325–2383, Oct. 1998.

[76] W. R. Gardner and B. D. Rao, "Theoretical analysis of the high-rate vector quantization of LPC parameters," *IEEE Trans. Speech and Audio Processing*, vol. 3, pp. 367–381, 1995.

[77] S. P. Lipshitz, M. Pocock, and J. Vanderkooy, "On the Audibility of Midrange Phase Distortion in Audio Systems ," *J. Audio Eng. Soc.*, vol. 30, pp. 580–595, Sept. 1982.

[78] M. R. Schroeder, "New results concerning monaural phase sensitivity," *J. Acoust. Soc. America*, vol. 82, pp. 1579, 1959, Oct. 1959.

[79] F. A. Bilson, "On the influence of the number and phase of harmonics on the perceptibility of the pitch of complex signals," *Acoustica*, vol. 28, pp. 60–65, 1973.

[80] R. Plomp and H. J. M. Steeneken, "Effect of phase on the timbre of complex tones," *J. Acoust. Soc. America*, vol. 46, pp. 409–421, 1969.

[81] A. E. Rosenberg, "Effect of glottal pulse shape on the quality of natural vowels," *J. Acoust. Soc. America*, vol. 49, no. 2, pp. 583–590, 1971.

[82] H. Pobloth and W. B. Kleijn, "On phase perception in speech." *Proc. IEEE Int. Conf. on Acoustics, Speech, Signal Processing* (Phoenix, Arizona), vol. 1, pp. 29–32, 1999.

[83] H. Fletcher, "Auditory patterns," *Rev. Modern Physics*, vol. 12, pp. 47–65, 1940.

[84] E. Zwicker and U. T. Zwicker, "Audio Engineering and Psychoacoustics. Matching Signals to the Final Receiver, the Human Auditory System ," *J. Audio Eng. Soc.*, vol. 39, pp. 115–126, Mar. 1991.

[85] D. Sen, D. Irving, and W. Holmes, "Use of an auditory model to improve speech coders," *Proc. IEEE Int. Conf. on Acoustics, Speech, Signal Processing* (Minneapolis), pp. II–411—II–414, 1993.

[86] E. Ambikairajah, A. G. Davis, and W. T. K. Wong, "Auditory masking and mpeg-1 audio compression," *J. Electronics & Comm. Eng.*, vol. 9, pp. 165–175, Aug. 1997.

[87] A. Sekey and B. Hanson, "Improved one-bark bandwidth auditory filter," *J. Acoust. Soc. Am.*, vol. 75, pp. 1902–1904, June 1984.

[88] *Basilar membrane.* International School for Advanced Studies, Italy, 1999. *http://www.sissa.it/bp/Cochlea/utils/basilar.htm.*

[89] M. Schroeder, B. Atal, and J. Hall, "Optimizing Speech Coders by Exploiting Masking Properties of the Human Ear," *J. Acoust. Soc. Am.*, vol. 66, pp. 1647–1652, June 1979.

[90] B. R. Glasberg and B. C. J. Moore, "Derivation of auditory filter shapes from notched-noise data," *J. of Speech and Hearing Research*, vol. 47, pp. 103–108, 1990.

[91] E. Zwicker and H. Fastl, *Psychoacoustics: Facts and Models.* Springer-Verlag, second update ed., 1999.

[92] M. R. Schroeder, B. S. Atal, and J. L. Hall, eds., *Objective measure of certain speech signal degradations based on masking properties of human auditory perception.* New York: Academic: Frontiers of Speech Communication, 1979.

[93] R. D. Patterson, "A pulse ribbon model of monaural phase perception," *J. Acoust. Soc. Am.*, vol. 82, no. 5, pp. 1560–1586, 1987.

[94] J. Lukasiak and I. Burnett, "SEW representation for low rate WI coding," *Proc. IEEE Int. Conf. on Acoustics, Speech, Signal Processing* (Salt Lake City, Utah), vol. 2, May 2001.

[95] W. M. Hartmann, *Signals, Sound, and Sensation.* NY: AIP Press, 1997.

[96] A. Ludwig, *Audibility of Phase Distortion.* Silicon Beach Communications Inc., USA, 1997. *http://www.silcom.com/~aludwig/Phase_audibility.htm.*

[97] D. A. Nelson and R. C. Bilger, "Octave masking in normal hearing listeners," *J. of Speech and Hearing Research*, vol. 17, pp. 223–251, June 1974.

[98] J. Ni and I. S. Burnett, "Waveform interpolation at bit rates above 2.4 kb/s," *TENCON '97. IEEE Region 10 Annual Conference. Speech and Image Technologies for Computing and Telecommunications., Proceedings of IEEE*, vol. 2, pp. 601–604, 1997.