CHAPTER 1

Principles of Speech Coding

W. Bastiaan Kleijn

Speech-Coding Research Dept.
AT&T Bell Laboratories
600 Mountain Ave. Murray Hill, NJ 07974, USA

Kuldip K. Paliwal

School of Microelectronic Engineering
Griffith University
Brisbane, QLD 4111, Australia

Contents

Speech Coding and Synthesis
Edited by W.B. Kleijn and K.K. Paliwal

1. Introduction

Speech has a central position in human communication and this is reflected in modern technology. Machines are commonly used to transmit, store, manipulate, recognize, and even create speech. For these operations, the speech signal is usually represented in a digital format. This format is relatively robust against distortion and it facilitates processing. An important attribute of a particular digital representation is its bit rate, which specifies how many bits are required to describe one second of speech.

When a digital speech signal is transmitted, the bandwidth required is a function of the bit rate. Similarly, when a digital signal is stored, the bit rate determines the space required on the storage medium. Thus, system cost is often a function of the bit rate of digital speech signal. The purpose of *speech coders* is to reduce the bit rate of digital speech signals. A speech coder always consists of an encoder and a decoder. The encoder takes the original digital speech signal and produces a low-rate bitstream. This bitstream is the input to the decoder, which constructs an approximation of the original signal.

This chapter provides an introduction to the methods used in speech coding. It is not intended to provide an in-depth history or overview of the field, but rather it is intended to provide insight into some of the commonly used methods. Technical details of some specific signal-processing techniques (such as the Levinson recursion) have been omitted, but the interested reader can consult the references for further information.

The chapter is organized as follows. Section 2 provides background information about speech coding and section 3 describes the basic attributes of speech coders. Section 4 discusses some of the properties of the speech signal, showing that the speech signal contains significant structure. Section 5 describes how linear prediction can be used to exploit this structure for speech coding. Section 6 provides a high-level overview of current speech coders, showing that they can be divided into two classes: waveform-approximating coders and parametric coders. This section forms the introduction for the following sections (7 and 8), which discuss these two coding classes in more detail. The chapter concludes with section 9, which discusses future trends expected in the area of speech coding.

2. Motivation for speech coding

The output of an analog-to-digital converter is often a linear pulse-code-modulated (PCM) signal. To guarantee good quality for telephone speech (bandlimited to 200-3400 Hz), the linear PCM signal must have a sampling rate of 8 kHz and a resolution of 16 bits/sample, resulting in a bit rate of 128 kb/s. This bit rate can be interpreted as a reference bit rate for uncoded speech.

The appropriate bit rate at which speech should be transmitted or stored depends on the cost of transmission or storage, the cost of coding (compressing) the digital speech signal, and the speech quality requirements. Before 1980, the high cost of

coding and the low speech quality meant that speech coding was used very little. A dramatic increase in the efficiency of digital signal processing hardware and recent advances in speech coding research have significantly changed this situation, and currently speech coding is used for a large number of applications.

In almost all speech coders, the reconstructed signal differs from the original signal. The bit rate is reduced by representing the speech signal (or parameters of a speech model) with reduced precision and by removing inherent redundancy from the speech signal. The original signal can be recovered exactly if redundancy only is removed (lossless coding), but if the parameter precision is reduced this is not the case (lossy coding) [1].

The process of representing a value or a vector with reduced precision is called quantization. The distortion in the reconstructed speech signal resulting from quantization is called quantization noise. Sometimes the speech model does not reproduce the speech signal accurately even when its parameters are not quantized. Coders using such models suffer from model-induced distortion. In the speech-coding literature, model-induced distortion is usually included in the term "quantization noise", and this broad meaning of the term "quantization noise" will be adopted in this chapter. In speech coding, the particular character of the distortion is very important. In some applications the primary goal is to make the reconstructed speech sound natural, while in others it is to maximize the perceived similarity of the original and reconstructed speech signals. What makes speech coding particularly challenging is that it is impossible to write these perception-based goals in the form of an objective criterion, which is a function of the original and reconstructed signals.

Perception plays a fundamental role in the art of speech coding. This can be illustrated with a simple example. In a linear PCM system, the quantization error is independent of the amplitude of the speech signal (assuming that the dynamic range is not exceeded). However, more quantization noise is perceived for signals of small amplitude than for signals of large amplitude. A louder signal *masks* the quantization noise. A simple method to exploit this masking effect is to quantize the signal samples on a logarithmic scale, rather than a linear scale. In a logarithmic scale, the step size between quantization levels becomes progressively larger with increasing amplitude. Eight-bit logarithmic quantizers (corresponding to a bit rate of 64 kb/s) are commonly used in network telephony in Europe, North America, and Japan [2]. Such simple logarithmic speech coders are generally an integral part of the analog-to-digital and digital-to-analog conversion processes.

Around 1980, the first practical digital speech coders appeared. One of the first applications was secure communication over a telephone network. For this application, the speech signal is first converted into a digital bitstream, which is then encrypted and transmitted over the telephone network using a modem. To allow wide network coverage with existing modem technology, a 2.4 kb/s coding standard, FS1015 [3], was introduced for this purpose. Because a very high priority was assigned to the capability of secure communication, a low speech quality and a cost of several tens of thousands of dollars per unit were considered acceptable. Low-cost speech coding was only possible at higher bit rates. In 1983, the CCITT

defined a 32 kb/s coding standard, G.721 [4], aimed at general telephone network applications (a revision of the standard is G.726). This coder provides high ("toll") quality at low cost.

The G.726 and FS1015 standards are examples of industry-wide standards, defined to allow interoperability of equipment at both ends of a communication system. If speech is coded for the purpose of being stored on a local device, no such standards are required. As a result, voice-storage systems often use proprietary technology which can be revised more frequently. An early example of such a proprietary voice-storage algorithm is the 16 kb/s subband coder [5, 6] which was used in AT&T private-branch-exchange (PBX) telephone switches. This algorithm provided medium speech quality at low cost.

Ten years after the definition of G.726 and FS1015, the cost-performance tradeoff had improved enormously. Speech coders could easily be implemented on a single low-cost digital signal processing (DSP) chip. As a result, speech coding has become an integral part of many communication systems. A particularly important application is mobile communication. Europe, North-America, and Japan each have different speech coding standards defined specifically for this purpose. These speech coders operate at medium bit rate (between 3 and 13 kb/s) and provide acceptable-to-good speech quality. Chapter 2 provides an overview of current speech-coding standards and those under development.

Applications which have a major economic impact, such as mobile communications, provide a strong incentive to increase the efficiency of speech coders. The next section explicitly identifies what coder attributes should be considered in the development or selection of a coder for a particular application.

3. Speech-coder attributes

In the previous section, it was seen that speech coders are often developed with a particular application in mind. To understand why coders differ for different applications, it is useful to discuss the speech-coder attributes which can be optimized for a particular application. The main attributes of a speech coder are:

- bit rate
- subjective speech quality
- computational complexity and memory requirements
- delay
- channel-error sensitivity
- signal bandwidth

The following subsections will briefly describe each of these attributes. (In addition to the attributes mentioned below, there are other attributes which may be important in particular speech-coding applications [7]. These include the capability of a speech coder to transmit nonspeech signals such as data, signaling and dial tones, and the capability to support speech and speaker recognition.)

3.1. Bit rate

The reduction in the bit rate of the bitstream is the primary motivation to use speech coding. Depending on system and design constraints, fixed-rate or variable-rate speech coders are used.

Most existing speech-coding standards describe fixed-rate coders. Fixed-rate coders are simpler to design than variable-rate coders because one does not have to define criteria which determine the bit rate of the coder for a particular time interval. For applications where a single, constant transmission channel (e.g. a conventional telephone connection) is available for communication, the bit rate cannot exceed a given value. In such applications, a fixed-rate speech coder transmitting at the highest feasible rate is the best solution. Fixed-rate coders with applications in secure telephony generally have low bit rates, 0.8 to 4.8 kb/s. Coders used for satellite and cellular telephony range from 3.3 to 13 kb/s, and coders intended for use in the general telephone network have bit rates of 16 kb/s and upwards.

Variable-rate coding is natural for many applications. For example, in mobile telephony using the code-division-multiple-access (CDMA) scheme [8], the bit rate of the individual users can be varied independently. The lower the *average* bit rate transmitted, the more users can be accommodated by the CDMA network. Nonreal-time applications, such as voice storage, are also good candidates for variable-rate coding. The simplest variable-rate systems have two coding modes, one for silence and background noise, and one for when speech is present. More sophisticated methods often have a larger number of coding modes and adjust the bit rate based on external factors such as remaining storage capacity, or network load. It is likely that in the future a large fraction of speech coders will be variable-rate coders. Variable-rate coders are discussed in more detail in chapter 7.

3.2. Speech quality

The quality of the reconstructed speech signal is a vital attribute of a speech coder. It is also a particularly problematic attribute because the evaluation of speech quality is a notoriously difficult problem. As yet, it has not been possible to find an objective criterion that correlates well with speech quality for a variety of speech coders and input signals. Furthermore, with decreasing bit rate, the quality of the reconstructed signal of coders becomes more and more dependent on the characteristics of the input signal, making it difficult to anticipate the behavior of a coder in real-world applications. Thus, extensive testing with human subjects is required before the suitability of a particular speech coder for a practical application can be judged. Multiple encodings, background noise conditions, and nonspeech sounds (e.g. music) are now routinely included in these tests. Despite such efforts, speech coders do not always perform as expected in the field [9].

Three measures are often used to assess the subjective quality of speech coders: the diagnostic rhyme test (DRT), the diagnostic acceptability measure (DAM), and the mean opinion score (MOS). The DRT measures intelligibility, whereas the

DAM provides a characterization of coded speech in terms of a broad range of distortions [10]. The MOS attempts to combine all aspects of quality in a single number and is, perhaps, the most commonly used measure for the subjective quality of the coded speech. The MOS is extracted from the results of a category-rating test performed by 20 to 60 untrained listeners. The listeners characterize each of a set of utterances with a score on a scale from 1 (for unacceptable quality) to 5 (for excellent quality). An MOS of 4.0 or higher defines good (or "toll") quality, which is nearly transparent. An MOS between 3.5 and 4.0 defines communication quality, which is sufficient for natural telephone communication. However, speech of communication quality does have clearly perceived distortion. At an MOS lower than 3.0, the reconstructed speech may be intelligible, but often lacks naturalness and speaker recognizability. Chapter 13 provides more detail on the assessment of speech quality for speech coding.

3.3. Complexity

The computational complexity and memory requirements of a speech coder determine the cost and power consumption of the hardware on which it is implemented. Except for a few applications, such as announcements and message transmission, the speech encoder has to operate in real-time. This means that the computational effort for a block of speech has to be performed within the duration of that block. Currently, the bit stream must be decoded in real-time in virtually all commercial applications.

To keep cost and power consumption low, speech-coding algorithms are usually required to run on one DSP chip. Thus, the computational requirements of speech-coding algorithms have tracked the performance of a single DSP chip. For mass-market applications (such as mobile telephones and answering machines) in which each user requires a separate speech-coding device, these chips are usually relatively inexpensive, fixed-point, 16-bit DSP chips. The main disadvantage of fixed-point DSP chips is that they are difficult to program. For that reason, floating-point 32-bit DSP chips are commonly used for real-time development systems. Floating-point chips are also sometimes used in applications where many users share a single hardware platform (such as a centralized storage system), and where unit cost is less important than development cost. In addition, speech coders are often implemented on multi-purpose platforms, which often contain floating-point chips (e.g. multimedia applications).

As integrated-circuit technology improves, the computational capabilities and storage capacity of DSP chips increase [11]. A speech coder which may be expensive to implement initially is far less expensive five years later. A fixed-point DSP chip from around 1980 could perform only 1 million instructions per second (MIPS) and contained 128 words of random-access memory (RAM) and 1024 (1K) words of read-only memory (ROM). A typical high-performance fixed-point chip produced in 1995 can perform 40 MIPS and has 4K words of RAM and 16K words of ROM. They also use significantly less power and are less expensive.

3.4. Delay

Delay is an issue which is of importance mainly for two-way communications. There are two thresholds for coder delay. The most basic threshold is the threshold where the delay begins to affect the dynamics of a conversation. For example, delays over 150 ms can be perceived as an impairment for certain highly-interactive conversational tasks. However, one-way delays as large as 400-500 ms can be tolerated in many situations without a significant reduction in overall performance for normal conversational tasks [12]. (ITU-T Recommendation G.114 provides guidance in this aspect.) The second threshold applies only if the presence of echo canceling equipment is not guaranteed. Network echoes can be objectionable even when the two-way delay is less than 100 ms. Because the network often imposes additional delays, strict delay limits have been imposed on coders aimed at applications where echoes may be present.

Coder delay is often divided into four separate components. The first of these is the *algorithmic delay*. Coders usually operate on a block-by-block by basis, each block being called a frame. One frame of data must be accumulated before processing can begin. Often the coder requires some additional look-ahead beyond the frame to be processed. The sum of the look-ahead and the frame length is the algorithmic delay. Second is the *computational delay*, which allows for the actual processing time required for the coder. In most implementations, the processing delay is equal to a frame length or slightly less, so that the processor is freed up for the next frame of speech data when it is available for processing. Next is the *multiplexing delay*. In many transmission systems, a block of bits corresponding to a frame is assembled by the encoder before it is transmitted. Similarly, at the receiver, the block of bits associated with a frame is usually assembled before decoding begins. The delay associated with these assembling operations is called the multiplexing delay. Finally, there is the *transmission delay*. The coder may be sharing a high speed transmission channel with other users, or it may be a dedicated channel. The transmission delay is usually, but not always, less than the frame size. It is often reasonable to assume that the multiplexing and transmission delays add up to one frame length if the application involves an unshared channel.

Using these components of the delay, it is possible to obtain a rough estimate of the overall one-way delay introduced by speech coders based only on the frame size on which they operate. The algorithmic delay is at least one frame; the computational delay is typically also one frame. Furthermore, the multiplexing and transmission delays add up to one frame. Thus, as a rule of thumb, most speech coders require at least three frames of one-way delay. However, many speech coders need significantly more because of the look-ahead.

The relation between frame size and delay can be illustrated by contrasting two coders with nearly the same bit rate. The ITU 16 kb/s standard, G.728, [13] was developed for situations where echoes are not always canceled: it has a frame size of 0.625 ms. In contrast, the 13 kb/s GSM standard [14] (used for mobile communications in western Europe), which was developed for situations where network echoes are cancelled, has a frame size of 20 ms. Chapter 6 provides an in-depth treatment

of coders developed for situations with strong constraints on delay.

3.5. Channel-error sensitivity

In many applications, the bit stream received at the far end of the channel is corrupted by channel errors. Thus, it is essential that the reconstructed speech signal does not suffer unduly from such channel errors. The types of channel errors that speech coders are supposed to handle are usually divided into two classes: random errors and burst errors. These two classes require different strategies to reduce their impact on the reconstructed speech signal.

Simple models are used to test performance for the two classes. For the random-error class, each transmitted bit has the same probability of error and the overall error rate is usually limited to between 1% and 5%. This type of model is used, for example, in the development of standards used in secure telephony (FS1015 [3] and FS1016 [15]). To counter random channel errors, the coder should provide reasonable output for a frame, even if a small proportion of the received information within that frame is incorrect. As will be discussed in chapters 9 and 10, robustness against such channel errors can be obtained by means of index-assignment algorithms and through proper quantizer design. In addition, error-correcting codes can be applied to all or a subset of the transmitted information.

In the burst-error class, error detection schemes are used to classify each frame of received bits as usable or not usable. If a frame of received bits is deemed unusable, the decoder enters into a special mode, which often means that the signal power is slowly abated, and the signal characteristic is made to converge slowly to a white noise signal.

3.6. Signal bandwidth

A speech signal can be band-limited to about 10 kHz without affecting its perception [16]. However, in telecommunications the speech-signal bandwidth is usually limited much more severely. The general telephone network limits the bandwidth of the speech signal to between 200 and 3400 Hz. This band-limitation results in the characteristic sound of telephone speech. Both the lower limit at 200 Hz and the upper limit at 3400 Hz affect the speech quality.

In most digital speech coders, the speech signal is sampled at 8 kHz, resulting in a maximum signal bandwidth of 4 kHz. In practice, the signal is usually band-limited to about 3600 Hz at the high-end. At the low-end the cut-off frequency is usually between 50 and 200 Hz. Although the ITU has defined a wideband speech coder (G.722, which has a bandwidth of 7 kHz and operates at 64, 56, and 48 kb/s), wideband speech coders are still not very common, mainly since speech coders usually interface with the telephone network with its narrowband characteristics. However, as is discussed in chapter 8, wideband speech coding has received increasingly more attention in recent years. Interestingly, the overall perceptual quality for a wideband

coder may be higher than a narrowband coder operating at the same bit rate [17]. This is despite the fact that a narrowband coder introduces less audible distortion to a narrowband signal than a wideband coder to a wideband signal. It is to be expected that wideband coders will be most common in applications such as video conferencing and videophones, which do not normally interface with the telephone network.

4. Properties of the speech signal

To build an effective speech coder, a good understanding of the properties of the speech signal and its perception is needed. Such an understanding leads to models which remove redundancy from the speech signal and which only parameterize perceptually relevant information.

While existing knowledge of the properties and perception of the speech signal is useful in the design of speech coders, it is far from complete. For example, it is currently not possible to construct a model of the vocal tract and vocal cords which provides natural sounding speech, and auditory models cannot determine the presence of reverberation in coded speech. The purpose of this section is to provide a basic overview of the known properties of the speech signal and its perception which are exploited in speech coding.

4.1. Time-domain and frequency-domain characteristics

The perception of speech is a complex process. It is not yet clear how the human auditory system processes the speech signal. However, it is known that both temporal and spectral analyses of the speech signal are performed [18]. This can be used as a justification for analyzing the speech signal in terms of its frequency-domain characteristics as well as its time-domain characteristics. To derive the frequency-domain properties, the speech signal is analyzed on a short-time basis using a window of 20 to 30 ms duration. This is done to deal with the nonstationary nature of the speech signal (it is reasonable to assume that the speech signal within such a window is stationary). In addition, to minimize spectral distortion, smooth windows (such as Hann, Hamming, etc. [19]) are used for short-time spectral analysis.

From here on, the term "power spectrum" will be assumed to reflect a short-time power spectrum computed over smooth windows of 20 to 30 ms in duration. Speech-signal segments are often characterized in terms of the properties of such power spectra. It is natural to separate two key attributes of the power spectrum. These are:

– the envelope of the power spectrum
– the fine-structure of the power spectrum.

The autocorrelation theorem [20] (sometimes also referred to as the Wiener-Khintchine theorem) shows that the autocorrelation function of a signal is the

inverse Fourier transform of its power spectrum. Because of the uncertainty relation, an inverse relationship exists between the autocorrelation and power-spectral domains: the fine-structure of the power spectrum corresponds to the "long-term" autocorrelation of the time-domain signal, and the power-spectral envelope corresponds to the "short-term" autocorrelation.

For most speech sounds, the envelope of the power spectrum is the main factor determining their linguistic interpretation. A particular shape of the envelope can be associated with a particular phoneme [21]. Microphones and filters of communications equipment often modify the power spectrum. In this respect, it is useful to recall that the characteristic sound of telephone speech results from the limited bandwidth.

For voiced segments of speech (e.g. vowels), the fine-structure of the power spectrum displays a harmonic structure. That is, sharp peaks in the power spectrum occur at regularly spaced frequency intervals of 75 to 400 Hz, the interval being dependent on the speaker and the utterance. The spacing between the harmonics is called the fundamental frequency. Generally, female speakers have a higher fundamental frequency than male speakers. According to basic signal-processing theory, it follows that a harmonic structure in the speech spectrum corresponds to a periodic time-domain signal. Thus, voiced speech segments have a nearly harmonic frequency-domain structure and a nearly periodic time-domain signal.

When the harmonic structure does not exist in the power spectrum, then the speech segment is called "unvoiced". In the time-domain such signal segments display a noise-like structure (no periodicity is apparent). Fricatives such as "f" are examples of unvoiced sounds. Whispered speech is completely unvoiced.

The structure observed in the speech signal reflects human physiology. Lung pressure forcing air through tensioned vocal cords results in oscillation of the vocal cords, and, as a result, periodicity of the air flow through the vocal tract. By varying the tension on the vocal cords, the frequency at which they oscillate (the fundamental frequency) can be varied. If the vocal cords do not oscillate, but air flow is present, then unvoiced speech sounds are generated. The air flow emanating from the vocal cords forms an excitation signal for the vocal tract. The excitation signal has varying levels of periodicity and determines the fine-structure of the power spectrum of the speech signal. The excitation signal has no distinctive spectral-envelope, other than a spectral tilt (which results from two different sources during the human speech production process: the glottal-wave shape during voiced speech contributes about -12 dB per octave to the spectral tilt [22], and the radiation of speech from the lips causes about +6 dB per octave). Other than the overall tilt, the spectral envelope is mainly determined by the shape of the vocal tract. By changing the shape of the vocal tract (for example, by moving the tongue, jaw, or lips), the spectral envelope is changed.

The relation between the time-domain signal and the power-spectrum is illustrated with a practical example in fig. 1 and fig. 2. Figure 1 shows the first part of the speech signal for the word "cheat" spoken by a female speaker. The first part of this word is an affricative sound, which begins as a stop or plosive, followed by an unvoiced fricative sound [21]. The stop release is clearly visible at the left of the

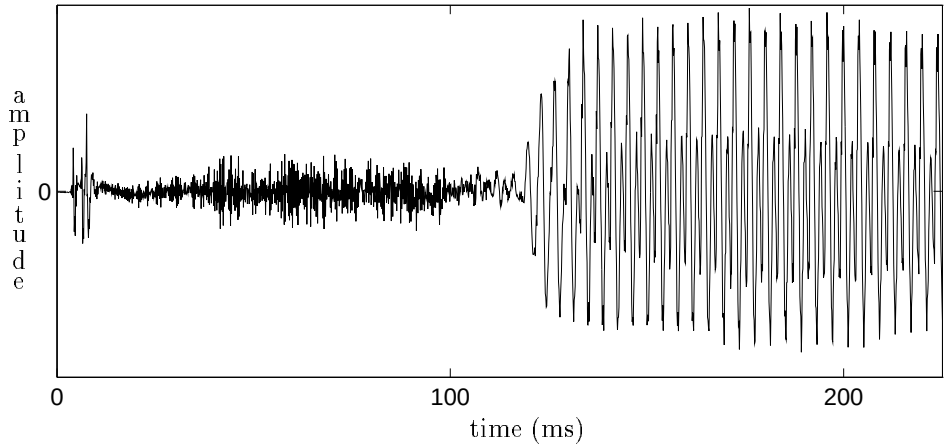

Figure 1. First part of the word "cheat" spoken by a female speaker.

figure. The unvoiced (nonperiodic) segment ends at 120 ms and is followed by a voiced (quasi-periodic) segment. The power spectra of fig. 2 show the spectra for 32 ms windows centered at 50 ms and 150 ms, respectively. The power spectrum of the periodic speech segment is easily interpreted as a harmonic fine-structure with a spectral envelope. The magnitude spectrum of the nonperiodic segment can be interpreted as the power spectrum of a white noise signal, shaped by a spectral envelope.

Much of the speech signal can be classified into voiced and unvoiced. However, many regions of natural speech display a combination of a harmonic spectrum and a noise spectrum. Generally, if the spectrum contains both harmonic and noise components, the harmonic components are more prominent at the lower frequencies, while the noise components are more prominent at the higher frequencies. A mixture of harmonic and noise components may appear over a large bandwidth. Speech coders which use a simple voiced-unvoiced classification often have difficulties in these regions.

Both the spectral envelope (short-term correlation) and the spectral fine-structure (long-term correlation) are relevant for speech coding. Short-term correlation implies that redundant information is contained in adjacent samples of a speech signal, which should be exploited in speech coding. In section 5, it will be shown how linear prediction can be used for this purpose. Because periodicity implies similarity between sequential cycles of the speech signal, an additional type of redundancy is found in the voiced speech signal. Such redundancy is not found in unvoiced signals, or, more generally, in the noise-like components of the speech signal. This means that reconstruction to within a certain signal-to-noise-ratio (SNR) requires a higher bit rate for unvoiced speech than for voiced speech.

Although the bit rate required to maintain a certain SNR is higher for unvoiced speech than it is for voiced speech, the bit rate required to maintain similar per-

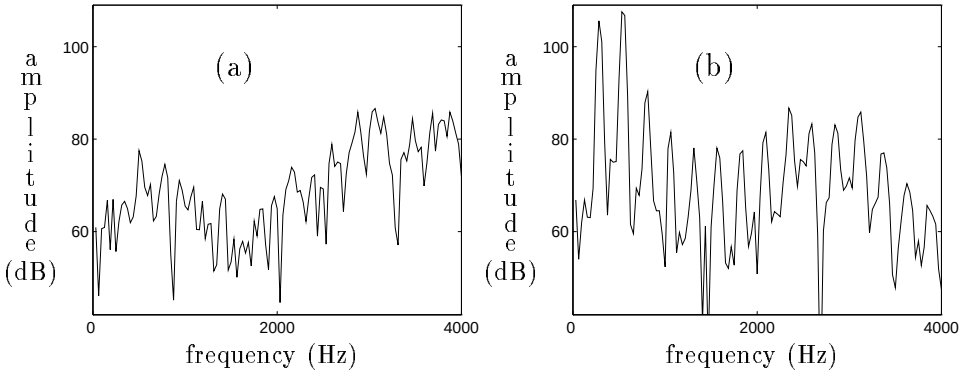

Figure 2. Speech spectra for the speech signal of fig. 1: (a) The power spectrum for a 32-ms nonperiodic (unvoiced) speech segment starting at 50 ms, (b) The power spectrum for a 32-ms periodic (voiced) speech segment starting at 150 ms.

ceptual quality is, in practice, lower for unvoiced than for voiced speech. If the unvoiced speech segments are replaced by a different signal with noise-like fine structure and similar spectral envelope, the perceived quality does not drop [23]. When the spectral envelope is identical, and the signal-power contour is correct, the human auditory system cannot distinguish between distinct noise-like signals. As a result, the bit rate required for perceptually accurate reconstruction of unvoiced signals is low, despite the high information content (in an information-theoretical sense) of the speech signal in these segments. In most current coding paradigms, this bit rate is lower than that required to obtain voiced-speech segments of similar perceptual quality.

The fore-mentioned qualitative assessments of the relative bit-rate requirements of various speech classes are based on practical experience with coding systems. No formal estimates of the information rate of different speech classes appear to exist. However, estimates of the average information rate do exist for the speech signal as a whole, and they are the topic of the next subsection.

4.2. Estimating the bit rate required for speech

Several different estimates can be made of the bit rate required for transmission of a speech signal. Obviously, the required bit rate depends on the amount of detail that is needed; i.e. it depends on the amount of coding distortion one is willing to accept. Because knowledge of both speech production and speech perception is limited, coders transmit more detail than necessary. However, it is possible to obtain some general estimates of the bit rate required, given particular sets of assumptions. In this subsection, three estimates, two given by Flanagan [22] and one provided by Johnston [24], will be discussed. All these estimates are based on information

theory [25], thus providing *average* required bit rates. The estimated rates are most relevant for variable-rate coders with long delays. The bit-rate requirements for constant-rate channels are higher.

A very low estimate of the required information rate is obtained if the speech signal is represented as a sequence of phonemes [22]. For simplicity, no correlation in the occurrence of adjacent phonemes is assumed to occur. An average information rate is then easily computed. According to information theory, the required average bit allocation per symbol is given by

$$I \;=\; -\sum_i P_i log_2(P_i), \tag{4.1}$$

where the labels P_i indicate the probability of a symbol i. English has about 42 phonemes (symbols), and it is normally spoken at a rate of about 10 phonemes per second. Using a table of relative probabilities of the phonemes [26], an information content of about 5 bits per phoneme can be computed, resulting in a total information rate of about 50 b/s. Note that natural silence is included in this bit rate. Even if information particular to a speaker (e.g. vocal-tract size and a description of the vocal cords) could be sent separately, a system transmitting only a sequence of phonemes lacks information about the particular pronunciation of an utterance. Thus, this estimate can be considered a lower limit for the bit rate required for a speech coder.

An upper estimate of the information rate of a speech signal can be obtained if its structure is ignored. The maximum rate at which information in a telephone-bandwidth signal with reasonable noise levels has to be transmitted, will be evaluated next [22]. Let P be the average power of the signal, W be the bandwidth of the signal, and G be the power of an additive noise signal. It is assumed that the additive noise signal is Gaussian white noise. An SNR of $P/G = 1000$ (30 dB) is assumed, as it corresponds to excellent speech quality. (The assumption of a fixed SNR ignores the information required for the transmission of the signal power, but in practice this is a small component of the overall bit rate.) Furthermore, let C denote the maximum information rate C which can be decoded from a signal with an arbitrarily small probability of decoding errors. Theorem 2 of the classic paper by Shannon [27] can be used to compute C for a signal containing the fore-mentioned additive noise:

$$C \;=\; W\ log_2(1 \;+\; \frac{P}{G}). \tag{4.2}$$

Thus, if the signal has a bandwidth of 3.5 kHz and an SNR of 30 dB, then it contains at most 35 kb/s of information. This is an estimate of the information rate required to describe a speech signal before it is processed by the human auditory system. However, in this estimate any structure (short-term correlation and long-term correlation) present in the speech signal is ignored. Structure in the signal implies redundancy which can be removed prior to transmission. As will be shown in the next section, linear prediction can be used to remove redundancy from the

speech signal, and an estimate of the reduction in the information rate will be provided in subsection 5.1.

When an acoustical signal is processed by the periphery of the human auditory system, its information rate decreases. Many features of an acoustic signal are not observed by the human auditory system due to masking. For example, a low-amplitude tone at a particular frequency (easily audible by itself) may be masked by a louder tone at a nearby frequency. Masking occurs in the frequency-domain as well as in the time-domain, but the former tends to be more important for coding purposes. (For more information on masking, see chapter 11 and [18].) It is relevant to consider the information rate of the signal after the removal of features which cannot be distinguished. This approach was taken by Johnston [24]. First, the complex Fourier transform of the speech signal is computed. The quantizer step size required to keep the quantization noise below the masking threshold is then measured for each of a set of bands. Both the masking thresholds and the bandwidths of the bands are based on knowledge of the human auditory system. The resulting estimate of the required information rate was called *perceptual entropy*. For telephone bandwidth speech, the perceptual entropy was estimated to be about 10 kb/s. This estimate is for sustained speech, and should correspond to the average bit rate required to perform transparent speech coding.

Transparent speech quality is not required for most coding applications. A less exacting goal is to reconstruct a natural speech signal which provides the same meaning and quality as the original speech signal. Ideally, the reconstructed signal should remain natural with decreasing bit rate, although the perceptual reproduction accuracy may decrease. Without reference to the original signal, it should not be possible to detect that the reconstructed speech signal was modified in low-bit-rate systems. However, in practice this is a difficult goal. The coding distortions are easily identified as not natural.

The discussion in this subsection shows that perception plays a major role in speech coding. It also shows that there is no obvious lower limit for the bit rate, other than a linguistically motivated rate of about 50 b/s. Coders producing acceptable quality speech currently operate at bit rates of 2 kb/s and higher, allowing plenty of scope to improve speech-coding performance in the future.

5. Redundancy removal and vocal-tract modeling through prediction

The previous section provided background information on the speech signal relevant for speech coding. To reduce the bit rate required for transmission of a speech signal, the structure that it exhibits must be exploited. This is often done by means of (linear) prediction techniques. Prediction can be used to remove redundancy from the speech signal or to create a model for the vocal tract. Because of the dominance of linear prediction (LP) methods in speech coding, it will be the subject of a fairly detailed description in this section.

When linear prediction is used in speech processing, its function can usually be interpreted in one of two distinct ways. The first view is that linear prediction is

used to remove redundancy from the speech signal. The removal of redundancy is performed with an LP filter, which has as output the prediction error. This filter is often called the LP analysis filter and the error signal is called the residual signal. The residual signal, which has a low level of redundancy, is particularly useful in speech coding, because it facilitates efficient encoding. In section 6 it will be shown that this first view is associated with what will be called waveform-approximating coders.

The second view is that the inverse LP filter, the LP synthesis filter, models the vocal tract, and that its transfer function describes the envelope of the speech signal. This interpretation is particularly relevant when LP analysis is used in speech recognition because it shows that the LP filter coefficients contain information relevant to classifying speech sounds. In speech coding, the second view is used in so-called parametric coders. The LP synthesis filter is used to describe the vocal tract, and an appropriate excitation for this filter is obtained from a model. The second view is also used in the design of quantizers for the LP filter coefficients. It is common practice to compare the power spectrum of the transfer function of the LP synthesis filter prior to and after quantization.

Section 5.1 explains how prediction removes the redundancy from the speech signal. This discussion is not limited to linear prediction, but also includes the case of nonlinear prediction. Section 5.2 then discusses the case of linear prediction in more detail, including the computation of the LP parameters. This is followed by sections which describe the relation to the power spectrum and the vocal tract.

5.1. Removal of redundancy through prediction

In digital speech signals, adjacent samples are often highly interdependent. This redundancy makes it more difficult to construct efficient quantizers for the speech signal. At the very least, vector quantizers would be required. However, the removal of redundancy from a signal usually means that a simpler quantizer can be used to encode the remainder signal. In the extreme case that all dependency is removed between signal samples, scalar quantization suffices.

In this subsection, it will be shown that a simple scalar quantizer is more effective for quantization of a speech signal after redundancy removal (by means of prediction) than for quantization of the speech signal itself. A scalar quantizer treats the signal samples as independent, and it was shown in subsection 4.2 that for a 30 dB SNR at least 35 kb/s are required to quantize such a signal. Below, an estimate will be provided for the bit rate required for a scalar quantizer applied to the speech signal after redundancy removal. This bit rate is significantly lower than 35 kb/s, providing a motivation to use prediction for speech coding.

Before the use of prediction for redundancy removal can be explained, the distinction between open-loop and closed-loop prediction must be discussed. The prediction is called open-loop when it is based on the original signal and closed-loop when it is based on the reconstructed signal. The prediction of an original-speech sample, $s(i)$, from a number of earlier original-speech samples, $s(i-1)$, $s(i-2)$, ..., $s(i-N)$,

can be denoted as

$$\tilde{s}(i) \;=\; P(s(i-1),\; s(i-2),\; \ldots\;,\; s(i-N)), \tag{5.1}$$

where $\tilde{s}(i)$ is the *open-loop* prediction for $s(i)$, $P(\cdot)$ is the predictor, and N is the predictor order. The open-loop residual is defined as

$$\tilde{e}(i) \;=\; s(i) \;-\; \tilde{s}(i). \tag{5.2}$$

A closed-loop residual signal is defined in the context of a reconstructed speech signal. This type of residual is obtained when the original speech signal is predicted from the earlier reconstructed speech samples. If the reconstructed signal is denoted as $\bar{s}(i)$, then closed-loop prediction is given by

$$\hat{s}(i) \;=\; P(\bar{s}(i-1),\; \bar{s}(i-2),\; \ldots\;,\; \bar{s}(i-N)). \tag{5.3}$$

The closed-loop residual is then defined as

$$e(i) \;=\; s(i) \;-\; \hat{s}(i). \tag{5.4}$$

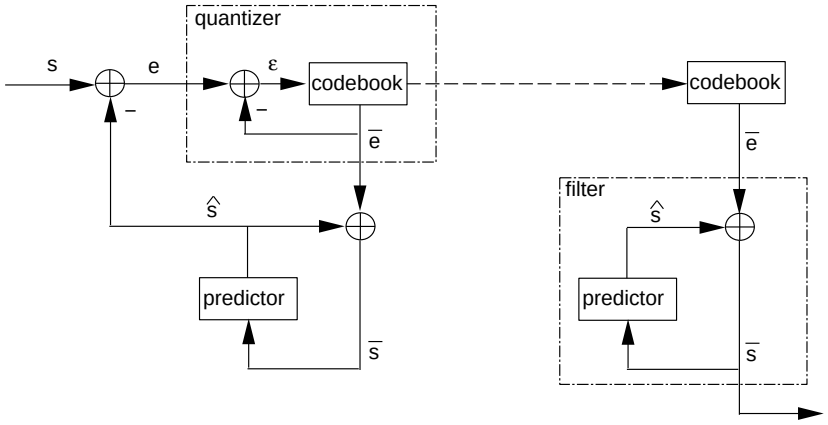

Figure 3. Block diagram of a prediction-based coder.

Figure 3 shows a coding structure for the closed-loop residual, which, in fact, is used in coders such as ITU G.726. For clarity, the time dependency has been omitted. The quantizer selects that value from a list of quantized values (the codebook) which is nearest in value to $e(i)$. This quantized value is denoted as $\bar{e}(i)$. An index

to this codebook value is transmitted to the receiver structure. The quantization process introduces a quantization error $\epsilon(i)$ given by

$$\epsilon(i) \;=\; e(i) - \bar{e}(i). \tag{5.5}$$

The (quantized) speech signal $\bar{s}(i)$ is reconstructed according to

$$\bar{s}(i) \;=\; \bar{e}(i) \;+\; P(\bar{s}(i-1),\; \bar{s}(i-2),\; \ldots,\; \bar{s}(i-N)). \tag{5.6}$$

Equation 5.6 can be seen as a filter with $\bar{e}(i)$ as input and $\bar{s}(i)$ as output, as is shown on the receiver side of fig. 3. The signal $\bar{e}(i)$ is the *excitation signal* for the filter.

Optimization of closed-loop prediction is equivalent to minimizing the power of the closed-loop residual signal $e(i)$. The closed-loop residual signal $e(i)$ generally has a smaller power than the speech signal $s(i)$. The ratio of the power of the speech signal and that of the closed-loop residual signal (averaged over a defined block length) is called the *closed-loop prediction gain*. While the signal power of $e(i)$ and $s(i)$ is different, comparison of eqs. 5.4 and 5.6 shows that a quantization error $e(i) - \bar{e}(i)$ gives rise to an identical error in the reconstructed speech signal $\bar{s}(i)$; i.e.,

$$e(i) - \bar{e}(i) = s(i) - \bar{s}(i). \tag{5.7}$$

Since the quantization error is proportional to the power of the signal to be quantized, and since the power of $e(i)$ is less than that of $s(i)$, it is advantageous to quantize $e(i)$ rather than $s(i)$.

Figure 4 demonstrates that the quantization error is the same in the speech domain and the residual domain. The structure in this figure is functionally identical to that of fig. 3. However, because the error in the speech domain is the one that matters, fig. 4 can be considered more fundamental than fig. 3. In particular, the structure of fig. 4 generalizes readily to the vector-excitation case.

The power of the reconstructed speech signal is larger than that of the excitation signal by a factor approximately equal to the prediction gain. However, as is shown in eq. 5.7, the quantization error of the reconstructed speech signal equals that of the excitation signal because of the closed-loop structure. The larger the prediction gain (the more redundancy is removed), the greater the advantage of quantizing the closed-loop residual signal rather than the speech signal itself. Direct quantization of the open-loop residual $\tilde{e}(i)$, in combination with the reconstruction according to eq. 5.6, is significantly less effective than closed-loop quantization since the quantization error is subject to the prediction gain in this case.

Ten dB is a typical value of the closed-loop prediction gain in a practical coding structure using short-term linear prediction. Equation 4.2 can then be used to obtain an estimate of the bit rate required to quantize the LP closed-loop residual such that the reconstructed speech signal has an SNR of 30 dB. Then, the SNR of the residual signal can be 20 dB. Thus, the average bit rate required to transmit

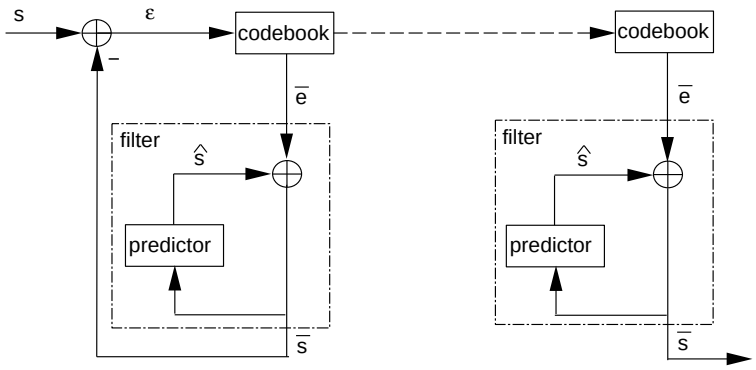

Figure 4. Modified block diagram of the prediction-based coder.

this information by means of a scalar quantizer is about 23 kb/s, significantly lower than the 35 kb/s required to transmit the speech signal itself.

The closed-loop prediction gain decreases with decreasing reconstruction accuracy. Also, the predictor is usually optimized for open-loop prediction, while a closed-loop structure is used for the quantization of the residual. The prediction gains quoted in the literature usually are open-loop gains. These open-loop gains provide an upper limit on the increase in coding efficiency resulting from the prediction process.

5.2. Linear prediction: definition and estimation

.

Equations 5.1 and 5.3 use a predictor $P(.)$ to predict the current speech sample from earlier N speech samples. These equations do not specify the type of predictor used; i.e., whether it is linear or nonlinear. If the samples of a digital speech signal are assumed to be Gaussian random variables, then *linear* prediction of a speech sample from earlier speech samples is optimal in the least-squares sense [28]. It is sometimes argued that the assumption of a Gaussian distribution of the speech signal (over short time segments) is accurate [2, 29]. On the other hand, if linear prediction is optimal for speech signals, then no further improvement should be expected from nonlinear prediction techniques. However, significant improvements in prediction gain have been shown [30–33]. As yet, such nonlinear prediction procedures have not led to commonly used coding schemes. The reader is referred to chapter 16 for more information on nonlinear processing of speech.

For a linear predictor, the open-loop prediction operation of eq. 5.1 can be written

as

$$\tilde{s}(i) \;=\; \sum_{n=1}^{n=N} \; -a_n \; s(i-n), \tag{5.8}$$

where $\{a_n\}$ are the linear-prediction (LP) coefficients. The open-loop LP residual is

$$\tilde{e}(i) \;=\; s(i) \;+\; \sum_{n=1}^{n=N} \; a_n \; s(i-n). \tag{5.9}$$

The summation in eq. 5.9 is usually interpreted as an LP error filter (a finite-impulse response filter).

Similarly, the closed-loop LP residual is

$$e(i) \;=\; s(i) \;+\; \sum_{n=1}^{n=N} \; a_n \; \bar{s}(i-n). \tag{5.10}$$

As shown in fig. 3, the closed-loop LP residual signal $e(i)$ is quantized to $\bar{e}(i)$ at the transmitting end, and at the receiver end it is used as an excitation signal to the LP synthesis filter to generate the reconstructed speech $\bar{s}(i)$; i.e.,

$$\bar{s}(i) \;=\; \bar{e}(i) \;-\; \sum_{n=1}^{n=N} \; a_n \; \bar{s}(i-n). \tag{5.11}$$

This operation is recognized as an all-pole filter.

Most speech coders employing linear-prediction use adaptation of the LP coefficients. This means that the coefficients must be estimated from a finite data sequence (typically 100 to 300 samples in speech coding). Generally, the predictor is optimized for open-loop prediction on a frame-by-frame basis. At normal levels of quantization accuracy, this will also result in good performance for closed-loop prediction. Optimization of the prediction coefficients for closed-loop prediction can lead to an improvement of performance, but requires an increased computational effort [34, 35].

Next, an estimation procedure for LP coefficients will be provided for a stationary, random signal with samples $s(i)$. First, the case where the statistics of the signal are assumed to be known will be described. Later, the more realistic case where only a sampled data sequence is known will be considered. The open-loop prediction residual for the LP case is given by eq. 5.9. The goal is to maximize the open-loop prediction gain, i.e. to minimize the expectation value of the open-loop prediction residual $\tilde{e}(i)$. It is convenient to define a vector of prediction coefficients, $\mathbf{a} \;\equiv\; [a_1 ... a_N]^T$, the autocorrelation $R(i-k) \;\equiv\; E[s(i)s(k)]$ ($E[]$ indicates expectation value), and the autocorrelation matrix $\mathbf{R}$, with elements $\mathbf{R}_{ik} \;\equiv\; R(i-k)$.

The autocorrelation matrix is symmetric and Toeplitz. From these definitions and eq. 5.9, it follows that the expectation value of the square of the prediction residual is

$$E[\tilde{e}(i)^2] = R(0) \; + \; 2 \sum_{n=1}^{n=N} a_n \; R(n) \; + \; \sum_{m=1}^{m=N} \sum_{n=1}^{n=N} a_m \; R(m-n) a_n$$

$$= \; R(0) \; + \; 2 \; [R(1) \; R(2) ... R(N)] \; \mathbf{a} \; + \; \mathbf{a}^T \mathbf{R} \mathbf{a}. \qquad (5.12)$$

The optimal vector $\mathbf{a}$ can be determined by taking the derivative of eq. 5.12 with respect to $\mathbf{a}$ and setting the result equal to zero:

$$\mathbf{a^T} \; \mathbf{R} \; = \; -[R(1) \; R(2) ... R(N)]. \qquad (5.13)$$

This equation is known as the *Yule-Walker equation* [36–38]. Because of the special structure of the Yule-Walker equation (the symmetric, Toeplitz structure of $\mathbf{R}$ and the particular form of the righthand side), it can be solved efficiently by means of the Levinson-Durbin recursion (e.g. [39, 38, 40]) or the Schur algorithm [41, 42].

A speech signal is not stationary and its statistics are not explicitly known. As mentioned before, however, it is standard practice to consider the signal as stationary over short (20 to 30 ms) time intervals. The predictor coefficients must be obtained from a short sequence of data samples describing such an interval. The *autocorrelation* method is the most popular method for estimation of the LP coefficients from such a sequence of data samples. In this procedure, the autocorrelations $R(i)$ are estimated from the original speech sample sequence and then inserted into the Yule-Walker equation. The use of the Yule-Walker equation means that the fore-mentioned fast methods can be used to solve for the predictor coefficients. In the autocorrelation method, the signal is first windowed with a smooth window (e.g. a Hamming window) of length L and with samples $w(i)$. The smooth nature of the window minimizes the distortion of the autocorrelation function, or, equivalently, the power spectrum. Consider the case where the window extends between $i = 0$ and $i = L - 1$. The autocorrelations are estimated as

$$R(k) \; = \; \sum_{i=0}^{i=L-1-k} w(i) s(i) \; w(i+k) s(i+k). \qquad (5.14)$$

These autocorrelations are used in eq. 5.13, and the resulting equation is solved to get LP coefficients $\{a_1, a_2, \ldots, a_N\}$ through the Levinson-Durbin algorithm or the Schur algorithm.

Note that the sum in eq. 5.14 is over less entries when the lag is increased. This introduces a bias in the autocorrelations, but it has as an advantage that the resulting filter is guaranteed to be minimum phase [38]. In practical terms, this latter property means that the (all-pole) LP synthesis filter used during speech reconstruction (see eq. 5.11) is guaranteed to be stable. While the above argument leads to the autocorrelation method in a very straightforward way, it can also

be shown that the autocorrelation method approximates a maximum-likelihood estimate of the predictor coefficients [36].

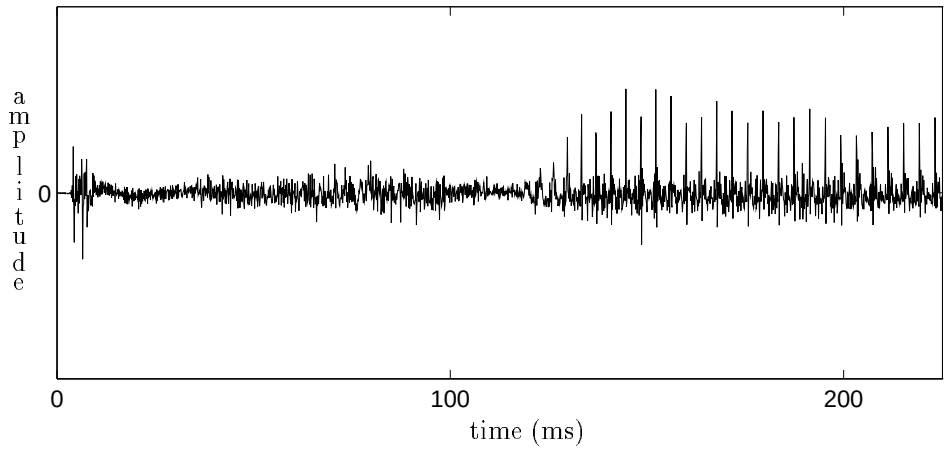

Figure 5. Segment of the LP residual of the word "cheat" spoken by a female speaker. The vertical scale is the same as in fig. 1

Figure 5 shows the residual signal obtained with an adaptive linear predictor for the same segment as shown in fig. 1. The predictor coefficients were computed every 20 ms using the autocorrelation method and interpolated for each 5 ms block. (The interpolation was performed on a transform of the LP coefficients, the line-spectral frequencies, which will be discussed later.)

Another well-known method for the computation of the predictor coefficients is the *covariance* method. In this method, the average of the squared prediction residual is minimized for a given, finite speech sample sequence. A disadvantage of the covariance method is that the resulting LP synthesis filter is not guaranteed to be minimum phase. Furthermore, the procedure requires more computational effort than the autocorrelation method. As a result, the covariance method is somewhat less common than the autocorrelation method; more information concerning this method can be found in references [38, 43].

5.3. Linear prediction: relation to the power spectrum

Linear prediction exploits the correlations in the speech signal. For a short-term linear predictor, the residual signal will have significantly less short-term correlation than the original signal. For a predictor of order N, the values of $R(1)$ through $R(N)$ are generally much smaller in amplitude for the residual signal than for the speech signal. From the autocorrelation theorem [20], it then follows that the power spectrum of the residual signal is significantly flatter than that of the speech signal. This is illustrated in fig. 6, which shows the power spectra for the residual signal corresponding to the power spectra of the speech signal shown in fig. 2.

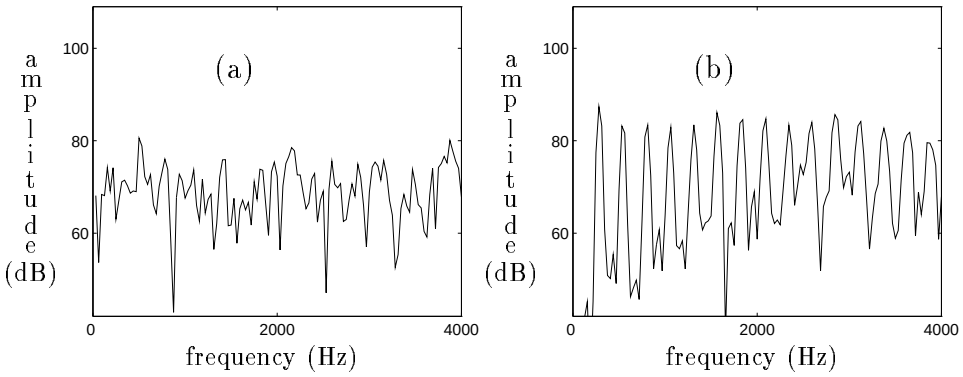

Figure 6. Spectra of the linear-prediction residual signal of fig. 5: (a) The power spectrum for a 32-ms window starting at 50 ms, (b) The power spectrum for a 32-ms window starting at 150 ms.

The filter structure at the receiver end of the predictive coder of fig. 3 has the excitation signal (the quantized residual signal) as input and the reconstructed speech signal as output. In the case of a linear predictor, this filter is a simple all-pole filter described by eq. 5.11, and its z-transform is given by

$$H(z) = \frac{1}{1 + a_1 z^{-1} + a_2 z^{-2} \cdots + a_N z^{-N}} = \frac{1}{A(z)}, \tag{5.15}$$

where $A(z)$ is the z-transform of the sequence $\{1, a_1, a_2, \ldots, a_N\}$. $A(z)$ represents the LP error filter. The all-pole filter $H(z)$ adds the spectral envelope missing from the excitation signal, resulting in the reconstructed speech signal. Because of this, it is also called the LP synthesis filter. It is completely defined in terms of the predictor coefficients $\{a_1, a_2, \ldots, a_N\}$ and contains information about the spectral envelope of the speech signal.

In order to understand the exact relationship between the original speech spectrum and the spectrum of the all-pole filter, let us define $\tilde{E}(z)$ as the z-transform of the residual signal $\tilde{e}(i)$ and $S(z)$ the z-transform of the original signal $s(i)$. The z-transform of eq. 5.9 is then

$$\tilde{E}(z) = A(z)S(z). \tag{5.16}$$

Let $f(\omega)$ denote the power spectrum of the signal; i.e.,

$$f(\omega) = |S(e^{j\omega})|^2, \tag{5.17}$$

and $\hat{f}(\omega)$ the power spectrum of the all-pole filter $H(z)$ (defined by eq. 5.15); i.e.,

$$\hat{f}(\omega) \;=\; \frac{1}{|A(e^{j\omega})|^2}. \tag{5.18}$$

From Parseval's theorem and eqs. 5.16, 5.17, and 5.18, it then follows that

$$\sum_{k=0}^{L+N-1} \{e(k)\}^2 \;=\; \frac{1}{2\pi}\int_{-\pi}^{\pi} \frac{f(\omega)}{\hat{f}(\omega)}\, d\omega. \tag{5.19}$$

Equation 5.19 shows that minimizing the residual signal is equivalent to minimizing the integral of the ratio of the power spectrum of the actual signal and the power spectrum of the all-pole filter. Thus, the power spectrum of the all-pole filter is indeed an approximation of the power spectrum of the original signal. Equation 5.19 shows that the errors in the power-spectrum estimate $\hat{f}(\omega)$ are weighted most severely where $f(\omega)$ is large. As an example, consider a signal consisting of sinusoids. The error criterion results in a power-spectrum estimate which is accurate at the location of the sinusoids (where $f(\omega)$ is large) but inaccurate in between, (where $f(\omega)$ is near zero), the actual value depends on the window. This type of power spectrum is closely approximated for voiced speech, as is seen in fig. 2(b).

5.4. Physiological basis for all-pole modeling

As was mentioned before, the properties of the speech signal reflect the physiological structure of the human speech-production system. Except for an overall tilt, the structure of the spectral envelope (or, equivalently, the short-term correlation) is determined by the vocal-tract shape. As a result, a model for the vocal tract can be used to describe the structure of the spectral envelope.

A simple model for the vocal tract can be obtained by making certain approximations and assumptions. The vocal tract is commonly approximated as a concatenation of rigid tubes, each of constant diameter. For frequencies which correspond to wavelengths which are large compared to the tube diameter, planar wave propagation can be assumed (this assumption is reasonable for frequencies below 4 kHz [39]). Furthermore, it is commonly assumed that losses due to viscosity and heat conduction can be neglected. It can then be shown that such a lossless tube model always results in an all-pole transfer function [39]. The commonly used conclusion is then that, since linear-prediction (eq. 5.9) leads to the all-pole transfer function (eq. 5.15), LP analysis (eqs. 5.14 and 5.13) can be used to estimate the parameters of this all-pole model (ignoring the tilt of the excitation signal).

Although this is not essential in the present context, a potential source of confusion should be pointed out. Under the fore-mentioned assumptions, the wave propagation through the vocal tract can be described by a set of simple linear equations [44] which include vocal-tract cross-sectional areas. Although these equations appear to be similar to a lattice filtering operation, the lattice filter and this

set of linear equations are equivalent only for particular terminations of the vocal tract [45, 36]. In general, it is not possible to obtain reliable estimates of the vocal-tract cross-sectional areas from an all-pole fit of the spectral envelope of the speech signal.

6. A classification of speech-coding procedures

Traditionally, speech coders have been separated into two classes: waveform coding and parametric coding. These classifications are generally used without clear definitions. The class distinction is further complicated by the introduction of the term *hybrid coders* for coders which fall into both of these ill-defined classes. (The term hybrid coders is usually reserved for the analysis-by-synthesis coders described in section 7.) To avoid confusion, the following two classes will be used in this chapter:

- *waveform-approximating coders*: coders which produce a reconstructed signal which converges towards the original signal with decreasing quantization error.
- *parametric coders*: coders which produce a reconstructed signal which does not converge to the original signal with decreasing quantization error.

6.1. The class of waveform-approximating coders

Waveform-approximating coders include those which are often known simply as waveform coders. In this class of coders, the objective is to minimize a criterion which measures the dissimilarity between the original and the reconstructed speech signals. In the most basic waveform-approximating coder, this criterion is the mean square difference between the original and the reconstructed signal, evaluated on a block-by-block basis. A block can be as small as one sample.

Because it does not consider masking phenomena in the human auditory system, the simple mean-square difference between the original and reconstructed speech is rarely used in present-day coders. Performance can be improved significantly by filtering the original and reconstructed speech signals with a filter which accounts for masking, prior to evaluating the criterion [46]. In addition, some recent coders exploit the fact that the human auditory system is insensitive to certain types of modification (such as minor time warps) of the speech signal. By selecting from a multitude of modified speech signals that signal which is encoded most efficiently, a further increase in efficiency is obtained [47]. This method is described in more detail in chapter 3.

In all waveform-approximating coders, the difference between two signals is minimized on a block-by-block basis. One of these signals is a processed version of the original signal and the other is a processed version of the reconstructed signal. Because of this structure, the coder output converges to the original signal as the bit rate increases.

Because the basic waveform is preserved in waveform-approximating coders, its SNR, expressed in dB, is generally positive. Measurement of the SNR is often used

as a sanity check during coder revisions. However, many new techniques that are proposed to obtain an increased perceptual quality are associated with a decrease in SNR.

The distortion criterion used for waveform-approximation coders naturally led to speech models which exploit the correlations observed in the speech signal. This is done by first removing the redundancy from the speech signal using prediction and then coding the closed-loop residual signal with lower accuracy. Early waveform-approximating coders primarily exploited short-term correlations in the speech signal, but more recent coders also exploit the long-term correlations.

6.2. The class of parametric coders

In parametric coding, the speech signal is characterized in terms of a set of model parameters, and these model parameters are quantized without consideration of the original speech signal. In these coders, measurement of the SNR is meaningless; the SNR is usually negative, when expressed as dB, and has no correlation with reconstructed-speech quality.

Note that a coder using a model for generating the reconstructed speech signal is not necessarily a parametric coder. Most waveform-approximating coders employ simple linear-prediction-based models for the speech signal. However, whereas waveform-approximating coders rely on an original signal to determine the quantized parameter values, this is not the case in a true parametric coder. As a result, the speech quality of parametric coders is limited by the accuracy of the model.

It seems natural and is common, to base the model of a parametric coder on the physiological structure of the human speech-production apparatus. Thus, it is usually possible to identify structures which emulate the vocal tract and structures which model the air flow emanating from the vocal cords. However, in more practical speech coders, the equivalence between the physiology and the model is often less important than speech-coder attributes such as computational effort and speech quality. Many models lump the spectral tilt observed in the speech signal with the vocal-tract structure. Models used in practical coders generally use a one-dimensional equivalent of the vocal tract.

It was already shown in section 5.4 that linear prediction can be used to construct a simple model of the vocal tract. Thus, many parametric coders use the same LP techniques that are used in waveform-approximating coders, despite the fact that the reasoning that led to parametric coders is very different from that which led to waveform-approximating coders.

7. Waveform-approximating coders

Waveform-approximating coders attempt to reconstruct the original signal waveform in a perceptually efficient manner. These coders generally use predictors to increase coding efficiency. This section will extend the general structure of the sim-

ple predictive coder discussed in subsection 5.2 in several ways. In particular, the use of vector quantization, perceptually motivated criteria, and subbands will be described.

7.1. Predictive coding

In subsection 5.1, it was shown how prediction can be used to remove redundancy from the speech signal. It was also shown that this redundancy removal is beneficial for speech coding because it allows simpler quantizers to be used.

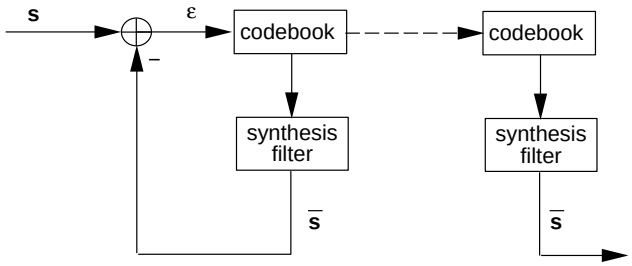

Figure 7. Block diagram of a basic predictive coder.

A high-level diagram of the predictive coding structure is shown in fig. 7 (this figure does not show the internal structure of the synthesis filter shown in fig. 4). For proper operation of the coder, the receiver must have access to the same synthesis filter as the transmitter. This can be done by keeping the predictor fixed, or by estimating it from earlier reconstructed speech, or by transmitting a synthesis-filter description as side information. The case where the predictor is transmitted as side information is known as *forward prediction*, whereas the case where the predictor is estimated from the reconstructed speech is known as *backward prediction*. Backward prediction is mainly used for higher bit rates (e.g. the 32 kb/s ITU G.726 coder). Because backward prediction is based on the earlier reconstructed speech signal, it tends to become less effective at low bit rates. As a result, forward prediction, which requires quantization of the prediction coefficients, is generally used for coders with lower bit rates. The quantization of the coefficients is described in subsection 7.1.4.

When linear prediction is used, as is usually the case, the coding structure of fig. 7 will be called *linear-prediction based analysis-by-synthesis (LPAS) coding*. For LPAS coders, the synthesis filter is the all-pole filter $H(z)$ of eq. 5.15. For a scalar predictive coder, each of a list of scalar values stored in a codebook is fed through the synthesis filter, using the same initial filter state. The index of the codebook entry which results in the smallest difference between the original and reconstructed speech signals is transmitted to the receiver. The sequence of selected codebook entries forms the excitation signal for the synthesis filter.

It is well-known that vector quantization is more efficient than scalar quantization [1]. This is especially true when there is correlation between the vector components, but it is true even when the vector components are uncorrelated. Coding efficiency is increased significantly by using vectors rather than scalars in the codebook for the structure shown in fig. 7. Because these coders are commonly used, a fairly complete overview of the procedure will be provided in the next few subsections. For a more detailed description, the reader is referred to chapter 3.

It must be emphasized that the meaning of "codebook" should be interpreted in a very wide sense in fig. 7. This means that scalar quantizers and single or multiple-stage codebooks are included. Thus, LPAS coders include adaptive-differential pulse-code modulation (ADPCM) coders, multi-pulse coders [48], regular-pulse coders [49], and code-excited linear prediction (CELP) coders [50]. The ADPCM and CELP coders form the best known group of LPAS coders. To illustrate how vector quantization is used in LPAS coders, a simple LPAS coder with a fixed vector codebook is discussed next.

7.1.1. An LPAS coder with a fixed vector codebook

The simplest vector-based LPAS coder uses a fixed vector codebook. It forms a good starting point because it can easily be generalized to more complex LPAS coders. Since the residual signal $e(i)$ has a large dynamic range, a gain-shape vector quantizer forms an efficient means for its quantization. In the following, vectors will be denoted by bold symbols (e.g. $\mathbf{e} \equiv [e(0)...e(M-1)]^T$) where M is the vector dimension. A candidate excitation vector describing a block of the excitation signal is then of the form

$$\bar{\mathbf{e}}_q = \lambda \mathbf{c}_q, \tag{7.1}$$

where λ is a scaling factor (or gain) and $\mathbf{c}_q$ is the codebook entry with index q, and the overbar again indicates that the signal is quantized. The winning codebook entry is selected because it results in the smallest difference with the original speech signal upon filtering with the LP synthesis filter $H(z)$ (eq. 5.15). Usually the size of the codebook is selected such that full use is made of an integer number of bits, i.e. the size is a power of 2. Common codebook sizes are from 128 to 1024 entries.

The difference between the original and synthesized speech signal must be evaluated on a block-by-block basis. In LPAS coders, these block sizes have varied from 5 to 80 samples. For optimal performance, the difference between the block of speech and each candidate reconstructed block of speech must be computed. In principle, each codebook entry $\mathbf{c}_q$ is filtered with the LP synthesis filter $H(z)$ (eq. 5.15), using the same zero-input response (i.e. the same filter memory). These computations are usually written in a vector notation as follows:

$$\bar{\mathbf{s}}_q = \lambda H \mathbf{c}_q + \bar{\mathbf{s}}_0, \tag{7.2}$$

where $\bar{\mathbf{s}}_q$ is the synthesized speech signal for codebook entry q, and $\bar{\mathbf{s}}_0$ is the zero-input response of the LP synthesis filter $H(z)$. The matrix H defines the filtering operation of the LP synthesis filter $H(z)$ with its memory set to zero. It is given by

$$H = \begin{bmatrix} h_0 & 0 & 0 & . & . & . & . \\ h_1 & h_0 & 0 & . & . & . & . \\ h_2 & h_1 & h_0 & . & . & . & . \\ . & . & . & . & . & . & . \\ . & . & . & . & . & . & . \\ . & . & . & . & . & h_0 & 0 \\ h_M & . & . & . & . & h_1 & h_0 \end{bmatrix}, \tag{7.3}$$

where $\{h_0, h_1, h_2, \cdots\}$ is the impulse response of the LP synthesis filter $H(z)$. Thus, $H\mathbf{c}_q$ in eq. 7.2 is the zero-state response of the LP synthesis filter $H(z)$ to vector $\mathbf{c}_q$.

To determine the optimal gain factor and codebook entry for a block of speech, the sum of the squares of the difference vector between the original signal and the synthesized signal must be minimized. The error criterion used is $(\mathbf{s}-\bar{\mathbf{s}}_q)^T(\mathbf{s}-\bar{\mathbf{s}}_q)$. By omitting the constant (and thus irrelevant) term $(\mathbf{s}-\bar{\mathbf{s}}_0)^T(\mathbf{s}-\bar{\mathbf{s}}_0)$, the error criterion η can be written as

$$\eta = \lambda^2 \mathbf{c}_q^T H^T H \mathbf{c}_q - 2\lambda \mathbf{c}_q^T H^T (\mathbf{s}-\bar{\mathbf{s}}_0). \tag{7.4}$$

Equation 7.4 can be used to determine both the codebook entry and the optimal gain factor λ. One method is to evaluate the criterion for all combinations of quantized λ values and all possible codebook entries $\mathbf{c}_q$.

7.1.2. The adaptive codebook and pitch prediction

As shown in fig. 1, the speech signal is highly periodic during voiced speech segments. Proper treatment of the periodicity is absolutely essential for good speech quality in low-bit-rate coding. The short-term linear prediction is usually based solely on correlations over intervals of less than 2 ms. As a result, the short-term linear predictor exploits correlations over intervals less than a pitch cycle, so it cannot describe the periodicity of the speech signal. This is why the residual signal shown in fig. 5 has the same level of periodicity as the speech signal shown in fig. 1: a separate coding structure is required to deal with the periodicity of the speech signal.

There are two different approaches to exploiting the periodicity of the residual (or speech) signal. Both are similar in implementation. In one approach, the periodicity is encoded using an *adaptive* codebook [51], which contains overlapping segments of the recent past of the LP excitation signal. Using this approach, an LPAS coder typically has both an adaptive codebook and a fixed codebook, which are searched sequentially. In the second approach (which appeared earlier histori-

cally), the periodicity is removed from the signal by means of closed-loop long-term prediction [52], and the remainder signal is encoded using the fixed codebook. For more details on this latter approach, the reader is referred to chapter 3.

When an adaptive codebook is used, the excitation signal consists of two contributions,

$$\bar{\mathbf{e}}_{pq} = \lambda^{(a)}\mathbf{c}_p^{(a)} + \lambda^{(f)}\mathbf{c}_q^{(f)}, \tag{7.5}$$

where $\lambda^{(a)}$ is the adaptive codebook gain, $\mathbf{c}_p^{(a)}$ is the adaptive codebook entry with index p, and $\lambda^{(f)}$ and $\mathbf{c}_q^{(f)}$ are the fixed-codebook gain and the fixed-codebook entry with index q, respectively. The adaptive codebook consists of segments of the recently synthesized excitation signal of length M:

$$\mathbf{c}_p^{(a)} = \bar{\mathbf{e}}(i - d(p)), \tag{7.6}$$

where $d(p)$ is a delay which specifies the start of the adaptive codebook vector with index p. Vectors for the case $d(p) < M$ must be treated separately [51].

The determination of the gains and indices of the adaptive and fixed codebook is performed in a sequential manner. First, eq. 7.4 is used to determine the adaptive-codebook contribution and gain. The selected adaptive-codebook contribution is then scaled and filtered through the LP synthesis filter $H(z)$ and subtracted from the original speech vector. Using this modified speech signal, the fixed codebook contribution is then determined, again using eq. 7.4.

In the original implementations of the pitch predictor, the delay $d(p)$ was constrained to be an integer number of samples. Later, it was realized that this is an artificial constraint [53], and implementations allowing fractional-delay implementations are now commonplace.

7.1.3. Perceptual weighting and postfiltering

The structure of fig. 7 can be further generalized by allowing operators which account for perceptual masking. (Masking is discussed in more detail in chapter 11.) Figure 8 shows such a structure. The identical perceptual-weighting operators (applied to original as well as synthesized speech) attempt to mimic the processing by the auditory periphery of the human auditory system. The matching error is then evaluated after this processing has been performed. Thus, the resulting error measure provides a more accurate description of the perceived quantization error.

Existing perceptual-weighting operators are generally linear. Linearity of the operator means that the signals $\mathbf{s}$ and $\bar{\mathbf{s}}$ can be added prior to the processing by the perceptual-weighting operator. This linearity assumption is implicit in many discussions about perceptual weighting.

Usually, the perceptual-weighting operator is a straightforward linear filter which accounts for spectral masking [46]. The impulse response of a linear perceptual-weighting filter can easily be included in the H matrix of eq. 7.3. A common form

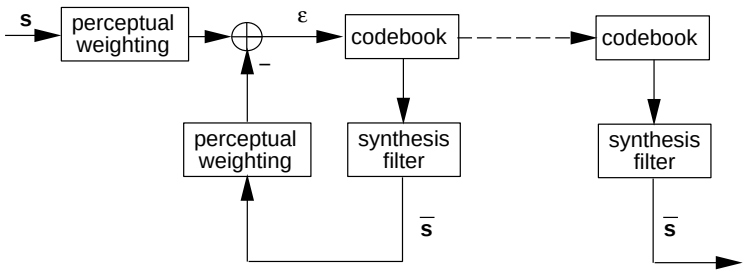

Figure 8. Block diagram of an LPAS coder with perceptual weighting and postfilter.

of this filter is [54]

$$w(z) \;=\; \frac{1-\sum_{i=1}^{i=N}\gamma_1^i a_i z^{-i}}{1-\sum_{i=1}^{i=N}\gamma_2^i a_i z^{-i}}. \tag{7.7}$$

In this equation the relative sizes of the weighting factors are: $1 \geq \gamma_1 \geq \gamma_2$. Typical values for the weighting parameters are 0.9 and 0.6, respectively [55]. The weighting filter deemphasizes the formant structure of the speech signal. This results in a larger matching error in the region of the formants, where spectral masking makes the auditory systems less sensitive to quantization errors.

A postfilter is usually applied to the reconstructed speech signal [54, 56]. This is illustrated in fig. 8. The postfilter is used to emphasize the formants of the speech signal. The emphasis is obtained with a pole-zero structure of the same form as the spectral weighting filter (eq. 7.7); typical values of the postfilter constants are $\gamma_1 = 0.5$ and $\gamma_2 = 0.75$ [55]. While it is desirable to emphasize the formant structure, the pole-zero filter (eq. 7.7) also reinforces the tilt of the reconstructed speech signal. Thus, the postfilter must include a tilt-control filter, which is usually a simple finite-impulse response filter [51, 56].

It should be noted that the postfilter emphasizes the formants of the speech signal. This means that quantization noise in the spectral valleys between the formants becomes less audible. It is important to note that quantization noise in the spectral valleys is also suppressed during the analysis stage with the spectral weighting filter of eq. 7.7. Thus, to obtain best performance, it is important to account for postfiltering in the determination of the spectral weighting parameters, γ_1 and γ_2, of the perceptual weighting filter [56].

7.1.4. The representation of linear-prediction coefficients

In LPAS coders which use forward prediction, information describing the LP coefficients is transmitted as side information. For efficient transmission of this information, the LP coefficients are subjected to quantization and interpolation. Interpo-

lation makes it possible to transmit the information about the LP coefficients less often (i.e., at a lower frame rate), thus reducing the bit rate.

However, both straightforward quantization and interpolation of the LP coefficients is problematic because small changes in the coefficients may result in large changes in the power spectrum, and, possibly, in unstable LP synthesis filters. Thus, quantization and interpolation are usually performed on transformed versions of the LP coefficients. A number of one-to-one mappings of the LP coefficients have been developed in attempts to find representations which minimize these shortcomings. These include the PARCOR coefficients and log-area ratios [36, 39], and the line-spectral frequencies (LSF) introduced by Itakura [57].

The most commonly used representation of the LP coefficients is the line-spectral frequencies (LSFs). While the LSF representation can be related to an acoustic tube model, this is rather artificial [58]. The real motivation for the wide-spread usage of LSFs is rather heuristic: it works well. To obtain the LSFs, the polynomial $A(z)$ is first decomposed into an odd and an even polynomial:

$$P(z) = A(z) \; + \; z^{-(N+1)} A(z^{-1}) \tag{7.8}$$

and

$$Q(z) = A(z) \; - \; z^{-(N+1)} A(z^{-1}). \tag{7.9}$$

It can be shown that, if $A(z)$ is minimum phase (i.e. it has its roots inside the unit circle), then the roots of both $P(z)$ and $Q(z)$ fall onto the unit circle, and the zeros of $P(z)$ and $Q(z)$ are interlaced [58]. These interlaced zeros of the $Q(z)$ and $P(z)$ polynomials are the LSFs. Thus, the LSFs can be found by sequentially searching for each successive LSF along the unit circle.

In practice, it is found that the LSFs behave very well when interpolated [59]. Many vector quantization procedures [1] have been used for quantization of the LSFs. It appears that for high quality coders, at least 20 bits are required to quantize the LSFs (assuming that no other information than a current window of speech data is used). Because single-stage quantizers of this size are not practical with current technology, suboptimal quantizers such as multi-stage VQ and split-VQ are used in practice. Split VQ methods divide the LSFs into two or three groups of adjacent LSFs [60] and have proven very effective. Chapter 12 provides a detailed description of the issues related to the quantization of the LP coefficients.

7.2. Subband coding

In most predictive coders, the full band of the speech signal is encoded. In contrast, in subband coding, the speech signal is first subdivided into a number of frequency bands, and each band is encoded separately. There are a number of advantages to this coding procedure [5]. Quantization noise is contained within a band, preventing the masking of one frequency range by quantization noise from another frequency

range. The dynamic range of the quantizers can be adjusted to the energy of a particular band, and the relative bit allocation for each band can be based on perceptual criteria.

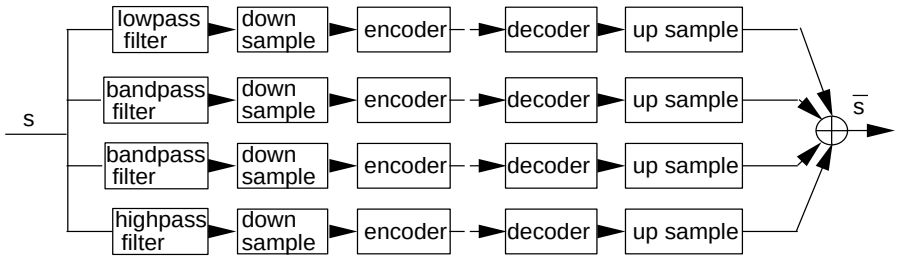

Figure 9. Block diagram of a subband coder with four bands.

Figure 9 shows the principle of subband coding. The full-band signal is first divided into a number of signals, each resulting from a different subband. Because of the reduced bandwidth of the subband signals, they can be downsampled prior to encoding. Each subband signal is then encoded, transmitted, and decoded. While earlier versions of subband coding do not use prediction [5], later versions use LPAS coders within each subband [61]. At the receiver, the subband signals are upsampled and added, rendering a reconstructed speech signal.

While subband predictive coders can provide substantial improvement over scalar LPAS coders, vector based LPAS coders have proven to be more successful than subband-based coders. As a result, relatively little recent work has focused on subband coders specifically aimed at speech. However, subband coding is used extensively in the coding of music, as will be seen in chapter 11.

7.3. Performance of waveform-approximating coders

LPAS coders are prevalent in speech coding between 5 and 32 kb/s. Compared to parametric coders, waveform-approximating coders tend to perform better with unexpected inputs such as music or various kinds of background noise. This is a result of the fact that waveform-approximating coders attempt to reconstruct the (perceptually weighted) input signal. Furthermore, most waveform-approximating coders do not require a long look-ahead buffer. The frame size is usually dictated only by the update of the LP coefficients. If the LP coefficients are computed from the earlier reconstructed speech (backward adaptation), then the delay of waveform-approximating coders can be made very low as is shown in chapter 6.

At first, it may be thought that subband coding has no significant advantages to offer over LPAS coders. After all, it can be argued that the bit allocation in LPAS coders is dynamic across the frequency band. The LP synthesis filter determines

in which regions of the spectrum the matching error is most significant. During the search through the fixed and adaptive codebooks, these spectral regions are emphasized. Implicitly, most bits are assigned to such a spectral region. Thus, the advantage of LPAS coders over subband coders is that this "dynamic bit assignment" is an integral part of the coder. However, LPAS coders have the disadvantage that their time-domain structure makes it difficult to introduce frequency dependent features. For example, it will be difficult to change their frequency resolution or time-resolution as functions of frequency. Such manipulations are much easier to accomplish in subband coders, and, maybe more important, in the parametric coders described in the next section.

8. Parametric coders

In section 6, a working definition for parametric coders was introduced: coders which produce a reconstructed signal which does not converge to the original signal with decreasing quantization error. In this section, the following three classes of parametric coders are discussed in some detail:

– linear-prediction based vocoders,
– sinusoidal-transform coders, and
– waveform-interpolation coders.

8.1. Linear-prediction based vocoders

As was mentioned in subsection 5.4, reasonable assumptions for the vocal tract lead to an all-pole model for the spectral envelope of the speech signal. The filter coefficients of the all-pole model can be estimated using LP analysis. This equivalency is exploited in linear-prediction based vocoders, where the LP synthesis filter is used as a model for the vocal tract (the spectral tilt contribution from the excitation is ignored). The excitation of the vocal tract is modeled as either a periodic pulse train (for voiced speech) or a white random number sequence (for unvoiced speech) [62, 3]. The speech signal is typically analyzed at a rate of 50 frames/s. For each frame the following parameters are transmitted in quantized form: *i*) the LP coefficients, *ii*) the signal power, and *iii*) the pitch period and voicing decision. The pitch period and voicing decision are usually transmitted with a 7-bit codeword, 127 of these codewords specify a particular pitch period, and the remaining codeword specifies the unvoiced mode.

Upon computation of the LP coefficients for a frame, pitch estimation is performed. In this procedure, the track of the pitch period (or its inverse, the fundamental frequency) is estimated for the frame, or the frame is declared unvoiced. Often the method estimates the pitch period at the frame boundary, and a continuous track is obtained by linear interpolation. Sometimes stepwise interpolation is used when pitch doubling or halving occurs, a phenomenon which occurs naturally

in speech. Many procedures for pitch estimation exist [63]. The simplest of these procedures are based on correlations.

An example of an effective correlation-based pitch-estimation procedure is described in reference [55]. In this procedure the correlation is

$$C(p) = \sum_{m=0}^{m=M-1} \check{s}(m)\check{s}(m-p), \tag{8.1}$$

where $\check{s}$ is the speech signal weighted with a filter as in eq. 7.7, with $\gamma_1 = 0.9$ and $\gamma_2 = 0.6$ (see eq. 7.7), and M is window length (80 in their implementation), all for an 8 kHz sampled signal. In general, it is found that it is good to use either a weighted speech signal or the low-pass filtered LP residual for pitch estimation.

It is important that the algorithm has a bias against finding multiples of the pitch period. To this purpose, the correlation function $C(p)$ is divided into three ranges of p: 20-39, 40-79, and 80-142. For each range the value p corresponding to the maximum for $C(p)$ is determined and normalized (division by $\sqrt{\sum_m \check{s}^2(m-p)}$). The normalized peak in the highest range is multiplied by a biasing value α^2 and the normalized peak in the middle range by α prior to selection of the final pitch period ($\alpha = 0.85$). For a more detailed description of pitch estimation procedures, see chapter 14.

The main weakness of the basic linear-prediction based vocoder is the binary decision between voiced and unvoiced speech. Such a binary voicing decision results in low performance for speech segments where both periodic and noisy frequency bands are present. By having a separate voicing decision for each of a number of frequency bands, the performance of these vocoders can be enhanced very significantly [64].

8.2. Sinusoidal coders

In sinusoidal coding, the speech spectrum is characterized by a sparse (complex or magnitude) spectrum. The sparse spectrum contains only the peaks of the complete spectrum. For synthesis, each point of the sparse spectrum is represented by a sine wave, whose amplitude and frequency are interpolated in time. This method is particularly natural when one considers the spectrum for voiced speech shown in fig. 2, which, in sinusoidal coding, would be described by its harmonics only. However, it should be emphasized that sparse spectra can also be used to represent unvoiced speech.

The types of parameters that are coded in sinusoidal coding differ strongly for different bit rates. At higher rates, the entire sparse spectrum (magnitudes, phases, and frequencies) and the overall power are transmitted. At lower bit rates, the phases are modeled and the frequencies are usually constrained to be harmonic. Thus, the fundamental frequency, the signal power, a description of the spectral envelope, and parameters for the phase model are transmitted at low bit rates.

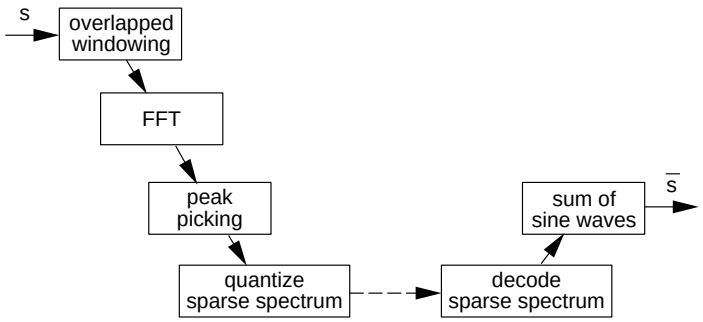

Figure 10. A sinusoidal coder

8.2.1. A basic sinusoidal coding system

The operation of a basic sinusoidal coder is shown in fig. 10. The speech signal is windowed with overlapping smooth windows. For an 8 kHz speech signal, a 256-point Hamming window is typically used. The complex spectrum for each of these windowed signals is obtained by means of a fast Fourier transform (FFT). The spectrum is separated into magnitude and phase spectra. The peaks in the magnitude spectrum are determined, and the rest of the spectrum is effectively set to zero. The magnitudes, the phases, and the frequencies of this sparse spectrum are quantized, and their quantization indices are transmitted to the decoder. At the decoder, the quantized sparse spectrum is reconstructed. The simplest procedure for getting speech from this sparse spectrum is to use the inverse Fourier transform. Since the windows used during analysis overlap, an overlap-add procedure must be used for synthesis [65]. Note that the overlap-add procedure avoids discontinuities in the reconstructed signal. More commonly used is reconstruction using a sum of evolving sine waves, as is indicated in fig. 10. This synthesis structure is often referred to as a *bank of oscillators*.

The sum of sine waves will now be discussed in more detail. A continuous time parameter t will be used, rather than the discrete time-sample index i used before. In the sum-of-sine-waves method, the reconstructed signal $\bar{s}(t)$ is obtained from the sparse spectrum by means of a sum of sine waves, each having a smoothly varying magnitude and phase function between the update times. Thus, the reconstruction is obtained from

$$\bar{s}(t) = \sum_{q=0}^{Q} A_q(t) sin(\phi_q(t)), \tag{8.2}$$

where $A_q(t)$ is the evolving magnitude of the sine wave with index q and $\phi_q(t)$ is its evolving phase [66]. Note that the instantaneous frequency of a particular sine

wave is the derivative of the phase: $\omega_q(t) = d\phi_q(t)/dt$.

The magnitude, phase, and frequency for each sine wave are specified by the sparse spectrum, which is transmitted at regularly spaced update times. The magnitude A_q of the sine wave can be interpolated linearly between the update points. Since the frequency ω_q of a sine wave q is the derivative of its phase, its phase track $\phi_q(t)$ must satisfy four boundary conditions: the phase and frequency at both endpoints. The lowest order polynomial function of t for the phase, $\phi_q(t)$, which can generally satisfy these conditions is a cubic polynomial [66]. The polynomial solution for which the phase is maximally smooth is selected. Usage of quadratic polynomials, which satisfy the boundary conditions only approximately, is also common [67].

All well-known sinusoidal coders separate the magnitude and phase spectra. The magnitude spectrum can be encoded in a number of different ways, including all-pole modeling, differential quantization, and other methods. For higher bit rates (e.g. 9.6 kb/s), the phase spectrum can be transmitted [65–67]. Excellent quality output can then be obtained with an analysis-synthesis system based on sinusoidal coding where sparse magnitude and phase spectra are updated once every 10 ms. At lower bit rates, the phase spectrum of the reconstructed signal is usually obtained by means of a model. Different models are used for voiced and unvoiced speech. Chapter 4, which describes sinusoidal coding, provides more detail on these models.

8.3. Waveform-interpolation coders

Voiced speech can be interpreted as a sequence of pitch-cycle waveforms. The general shape of these pitch-cycle waveforms evolves slowly as a function of time, facilitating prediction and interpolation. In waveform interpolation coding, a *characteristic waveform* is extracted at regular intervals [68]. These waveforms are placed along an axis perpendicular to the time axis. Thus, a two-dimensional signal $u(t, \phi)$ is obtained, which shows the characteristic waveform along the ϕ axis and the evolution of this waveform along the t (time) axis.

Waveform interpolation is usually applied to the LP residual. As will be shown in chapter 5, the motivation for using the residual signal rather than the speech signal is based on implementation issues. If the LP residual is used, then the transmitted parameters in a simple waveform interpolation coder are: the pitch period, the characteristic-waveform shape, the signal power, and the LP coefficients. For more recent coders, two waveforms are transmitted, one representing the noise component and the other the periodic component of the speech signal.

8.3.1. A basic waveform interpolation system

Figure 11 shows the basic operations which are present in a waveform-interpolation coder. Like most coders, waveform interpolation coders operate on a frame-by-frame basis. For each frame, the pitch track is determined first, using procedures similar to those used in linear-prediction based vocoders. Next, the characteristic waveforms

are extracted, at rates which vary from 40 to 500 Hz between different coders.

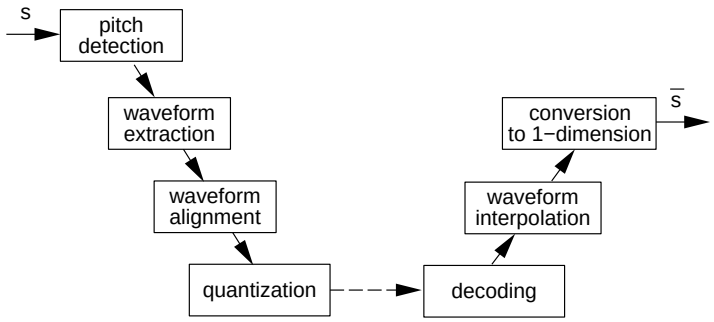

Figure 11. A waveform-interpolation coder

In general, the features of successively extracted waveforms (e.g. pitch pulses) are not aligned. An alignment procedure brings out the similarity of the sequence of extracted waveforms. For the alignment procedure, it is usually assumed that the characteristic waveform $u(t,\phi)$ at any particular time t is periodic in ϕ. As a result, the main features of successively extracted waveforms can be aligned by means of circular precession of the waveform. The succession of extracted, aligned waveforms defines the surface $u(t,\phi)$. This surface is shown in fig. 12 for the speech segment shown in fig. 1. For the voiced speech segment, the characteristic waveform evolves slowly, while for the unvoiced segment, the characteristic waveform evolves rapidly.

If the characteristic waveform is considered to be one cycle of a periodic signal in ϕ, then a Fourier-series is a natural representation. The evolving characteristic waveform can then be described as

$$u(t,\phi) = \sum_{q=1}^{q=Q} \alpha_q(t) sin(q\phi) + \beta_q(t) cos(q\phi). \tag{8.3}$$

The Fourier-series is not the only representation for the characteristic waveform. Other representations are being used as well.

Continuing the operations in fig. 11, the next step is quantization of the coder parameters, which usually are the pitch period, the LP coefficients, and a description of the characteristic waveform. At the receiver, these parameters are decoded, resulting in a characteristic waveform surface $\hat{u}(t,\phi)$ sampled in time t. Using interpolation, the surface is upsampled to one characteristic waveform per output sample. This interpolation lends its name to the procedure. Finally, a one-dimensional output signal is reconstructed from the two-dimensional evolving-waveform signal by specifying the phase $\phi(t)$ for each sample of the output signal,

$$\bar{s}(t) = \hat{u}(t,\phi(t)). \tag{8.4}$$

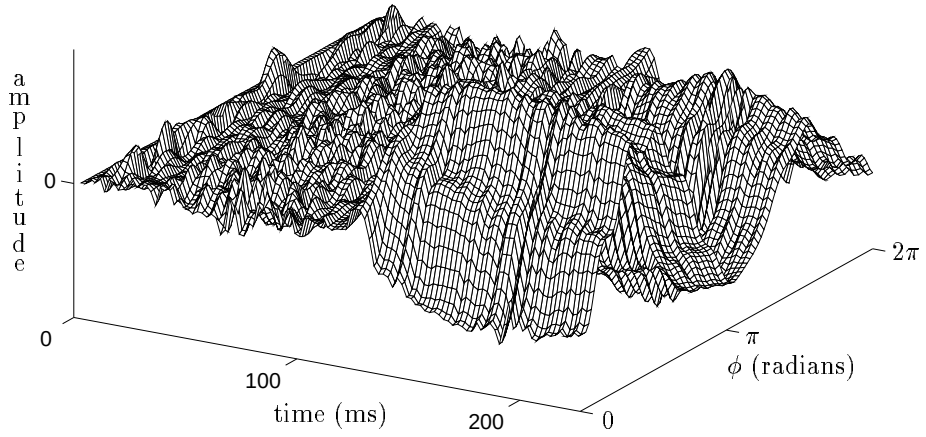

Figure 12. The evolving characteristic waveform for the same segment of the word "cheat" as fig. 1. The signal is oversampled along both the time and ϕ axis.

The phase ϕ is the integral of the fundamental frequency ω_0 (the inverse of the pitch period),

$$\phi(t) = \int_0^t \omega_0(t)dt. \tag{8.5}$$

The fundamental frequency is interpolated between updates.

Usually waveform interpolation is applied to voiced speech only, and the characteristic waveform represents a pitch-cycle waveform. In recent implementations, the method is also used for unvoiced speech [69]. For unvoiced speech there is no correlation between waveform segments which do not overlap. This contrasts with voiced speech where such segments are highly correlated even when separated by multiple pitch periods. Thus, the characteristic waveform evolves rapidly during unvoiced speech and slowly during voiced speech, as is illustrated in fig. 12. More generally, the voiced component of the characteristic waveform evolves slowly, and the unvoiced component of the characteristic waveform evolves rapidly. The level of periodicity in the one-dimensional signal has been replaced by the rate of evolution in the two-dimensional characteristic waveform signal. This means that the voiced and unvoiced components can be separated by a simple, nonadaptive filtering operation along the t axis.

It is this separation of the characteristic waveform into a slowly-evolving waveform (representing the voiced component) and a rapidly-evolving waveform (repre-

senting the unvoiced component), which allows waveform interpolation to be used for both voiced and unvoiced speech. The slowly-evolving waveform can be downsampled to a low rate (e.g. 40 Hz), allowing a low bit rate while maintaining a high level of accuracy. For the coding of the rapidly evolving component, it is useful to recall that for the unvoiced speech component only the signal-power envelope and the spectral envelope carry perceptually significant information [23]. Thus, the rapidly-evolving waveform can be encoded at a low rate using a rough description of the magnitude spectrum and little or no information for the phase spectrum.

8.4. Differences between waveform interpolation and sinusoidal coding

When the Fourier-series representation is used, waveform-interpolation coding has similarities to sinusoidal coding. However, there are a number of significant distinctions. At the analyzer, the main distinction is the analysis window of waveform interpolation (which adapts with the pitch period) and the high extraction rate. The short window length used in waveform interpolation means that the complex spectrum defined by the Fourier series does not resolve the individual harmonics. In fact, because an extraction window of length equal to one pitch period is used, each term of the Fourier series of eq. 8.3 specifies the complex amplitude of a harmonic. The selection of the appropriate pitch period means that peak-picking is not relevant for the waveform interpolation method. Conversely, the sinusoidal coding method has no operation corresponding to the alignment procedure, which maximizes the similarity between successively extracted characteristic waveforms in waveform interpolation.

At low bit rates, both sinusoidal coding and waveform interpolation distinguish between a noise component and periodic component of the speech signal. However, whereas in sinusoidal coding this distinction is based on the properties of a single spectrum, in waveform interpolation it is obtained by decomposition of the evolving waveforms by means of filtering in the time direction.

The procedures used for speech synthesis also differ for the sinusoidal coder and the waveform interpolation using a Fourier-series representation. Generally the interpolation interval in waveform interpolation coders is shorter than what is used in sinusoidal coding. The basic interpolation mechanisms differ as the result of the different perspectives; both interpolation mechanisms can be used for either type of coder.

In the waveform-interpolation procedure, the Fourier-series coefficients α_q and β_q are interpolated linearly in time to render a complete definition of the surface $u(t, \phi)$. The phase is a quadratic polynomial of time (if ω_0 is linearly interpolated), as is seen from eq. 8.5. Thus, it follows that the boundary conditions used in waveform interpolation are different from those commonly used in sinusoidal coding. In waveform interpolation, the Fourier-series coefficients (i.e. the complex spectrum) and the fundamental frequency are specified at the update points, whereas in sinusoidal coding the magnitude, phase, and frequency are specified for each sinusoid at the update points. Sinusoidal coding employs a different cubic polynomial for the

phase of each harmonic, while waveform interpolation uses only one quadratic phase polynomial for all harmonics. In waveform interpolation the complex spectrum is interpolated, while in sinusoidal coding the magnitude spectrum is interpolated. Thus, two terms [a $sin(.)$ and a $cos(.)$] are needed for each harmonic in waveform interpolation, whereas sinusoidal coding requires just a $sin(.)$ function for each harmonic. However, the regular phase structure in waveform interpolation makes it amenable to fast synthesis procedures.

8.5. Performance of parametric coders

Parametric coders have traditionally been used for bit rates below 5 kb/s. Above these rates, it is difficult to reach the quality levels established by waveform-approximating coders with a parametric coder. With the increasing sophistication of parametric models, however, parametric coders may find applications at higher rates. The likelihood of such applications is motivated by the fact that the reconstructed speech quality for some parametric coders is very high when the parameters are not quantized. This means that model-induced distortion is low. Yet, because of computational complexity and relatively low robustness against background noise, it is to be expected that the main range of bit rates where parametric coders are applied will remain below 5 kb/s. In 1994 the speech quality produced by parametric coders between 2.4 and 5 kb/s had reached a level where many commercial applications, such as mobile communications, had become viable.

In practice, the best of each of the three types of parametric coders discussed in this section currently perform similarly in comparative testing. Because parametric coders are generally optimized to code speech under a specific set of conditions, their performance can be erratic for nonspeech sounds. For example, a classification into voiced and unvoiced speech may switch between these two classes in a steady-state background-noise signal, resulting in an annoyingly nonsteady output signal. All low-rate parametric coders perform poorly for music, which often has a complex spectral structure.

9. Future trends

Since 1980, very significant progress has been made in the field of speech coding. This progress resulted from *i*) improved understanding of the structure of the speech signal, *ii*) improved understanding of the human auditory system, *iii*) better quantization techniques, and *iv*) much faster signal-processing hardware. These trends are likely to continue, at least in the near future.

It does seem likely that there will be a shift in the focus of various speech-coding research groups. From about 1985 onward, much of the progress in speech coding was the result of work based on the LP based analysis-by-synthesis (LPAS) coding paradigm. Numerous standards between 3 kb/s and 16 kb/s have been based on this paradigm, and the LPAS coder is likely to remain the dominant paradigm for bit

rates between 5 and 16 kb/s for many years to come. However, it now appears that if the bit rate is reduced below 5 kb/s, then the advantage of the LPAS paradigm over other methods becomes progressively less. This is illustrated by the fact that no LPAS coders were entered in a recent coding survey of 2.4 kb/s coders [70]. With increasing economic pressure to reduce the bit rate of coders, it is likely that a major focus of research in the coming years will be on one or more parametric coding procedures, including the methods described in section 8.

Many of the new applications of speech coders, such as cellular telephony and answering machines, do not require a fixed bit rate. As a result, a significant effort has been dedicated to variable-rate coding in recent years. Because of the increase in the number of applications for these coders, it is likely that there will be a significant increase in research on variable-rate coders.

The minimum bit rate that speech coders will achieve is limited by the information content of the speech signal. As was shown in section 4.2, what that information content is depends on what one attempts to measure. But for a lower limit, the critical information rate is the rate obtained after the signal is interpreted by a person. Of course, it is notoriously difficult to evaluate this information rate quantitatively. Even worse, certain information about the speech which is noticeable by humans can be changed without affecting the usefulness of a coder. For example, a speaker who reads a paragraph twice generates two very different waveforms, but for many cases these differences will not affect the interpretation by the listener. Thus, it is very difficult to say how far the bit rate of speech coders will ultimately fall. Nevertheless, it is reasonable to assume that the perceptually relevant information rate is within an order of magnitude of 100 b/s. This suggests that, whereas in the last 10 years bit rates (for a given speech quality) have fallen by approximately an order of magnitude, it is unlikely that this feat can be repeated in the next 10 years.

References

[1] A. Gersho and R. M. Gray, *Vector Quantization and Signal Compression*. Dordrecht, Holland: Kluwer Academic Publishers, 1991.

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

[3] T. E. Tremain, "The government standard linear predictive coding algorithm," *Speech Technology*, pp. 40–49, April 1982.

[4] CCITT Study Group XVIII, "Temporary document 18, draft recommendation, g.78zz," November 1983.

[5] R. E. Crochiere, S. A. Webber, and J. L. Flanagan, "Digital coding speech in sub-bands," *Bell Syst. Techn. J.*, vol. 55, no. 8, pp. 1069–1085, 1976.

[6] J. Josenhans, J. Lynch, M. Rogers, R. Rosinski, and W. VanDame, "Speech processing application standards," *Bell Syst. Techn. J.*, vol. 65, no. 5, pp. 23–33, 1986.

[7] S. Dimolitsas, C. Ravishankar, and G. Schroeder, "Current objectives in 4-kb/s wireline-quality speech coding standardization," *IEEE Signal Processing Letters*, vol. 1, no. 11, pp. 157–159, 1994.

[8] R. Pickholtz, L. Milstein, and D. Schilling, "Spread spectrum for mobile communications," *IEEE Trans. Vehic. Techn.*, vol. 40, no. 2, pp. 313–322, 1991.

[9] T. Wigren, A. Bergstrom, S. Harrysson, F. Jansson, and H. Nilsson, "Improvement of background sound coding in linear predictive speech coders," in *Proc. Int. Conf. Acoust. Speech Sign. Process.*, (Detroit), 1995.

[10] W. Voiers, "Diagnostic acceptability measure for speech communications system," in *Proc. IEEE Int. Conf. Acoust., Speech, Sign. Process.*, pp. 204–207, 1977.

[11] J. Boddie, "Digital signal processor," *Bell System Technical Journal*, vol. 60, no. 7, pp. 1431–1439, 1981.

[12] S. Dimolitsas and J. Phipps, "Experimental quantification of voice transmission quality of mobile-satellite personal communications systems," *IEEE Journal on Selected Areas in Comm.*, vol. 13, no. 2, 1995.

[13] J.-H. Chen and M. S. Rauchwerk, "An 8 kb/s low-delay CELP speech coding algorithm," in *Proc. IEEE Global Telecomm. Conf.*, (Phoenix), pp. 1894–1897, 1991.

[14] P. Vary, R. Hoffman, K. Hellwig, and R. Sluyter, "A regular-pulse excited linear predictive code," *Speech Comm.*, vol. 7, no. 2, pp. 209–215, 1988.

[15] J. P. Campbell, V. C. Welch, and T. Tremain, "The DOD 4.8 kbps standard (proposed federal standard 1016)," in *Advances in Speech Coding* (B. S. Atal, V. Cuperman, and A. Gersho, eds.), pp. 121–133, Dordrecht, Holland: Kluwer Academic Publishers, 1991.

[16] P. Denes and E. Pinson, *The Speech Chain.* Garden City, New York: Anchor Books, 1963.

[17] G. Roy and P. Kabal, "Wideband CELP speech coding at 16 kbits/s," in *Proc. Int. Conf. Acoust. Speech Sign. Process.*, (Toronto), pp. 17–20, 1991.

[18] B. C. J. Moore, *An Introduction to the Psychology of Hearing.* London: Academic Press, 1989.

[19] A. V. Oppenheim and R. W. Schafer, *Digital Signal Processing.* Englewood Cliffs, NJ: Prentice-Hall, 1975.

[20] R. Bracewell, *The Fourier Transform at Its Applications.* New York: McGraw Hill, 1986,.

[21] H. Edwards, *Applied Phonetics.* San Diego: Singular Publishing Group, 1992.

[22] J. L. Flanagan, *Speech Analysis Synthesis and Perception.* Berlin: Springer-Verlag, 1972.

[23] G. Kubin, B. S. Atal, and W. B. Kleijn, "Performance of noise excitation for unvoiced speech," in *Proc. IEEE Workshop on Speech Coding for Telecomm.*, (Sainte-Adele, Quebec), pp. 35–36, 1993.

[24] J. D. Johnston, "Estimation of perceptual entropy using noise masking criteria," in *Proc. Int. Conf. Acoust. Speech Sign. Process.*, (New York), pp. 2524–2527, 1988.

[25] L. Brillouin, *Science and Information Theory.* New York: Academic Press, 1962.

[26] G. Dewey and d. Cambridge, MA, *Relative Frequency of English Speech Sounds.* Harvard University Press, 1923.

[27] C. E. Shannon, "Communication in the presence of noise," *Proc. I.R.E.*, vol. 37, pp. 10–21, 1949.

[28] T. Kailath, *Lectures on Wiener and Kalman Filtering.* New York: Springer Verlag, 1981.

[29] B. S. Atal, "Predictive coding of speech signals at low bit rates," *IEEE Trans. Comm.*, vol. COM-30, no. 4, pp. 600–614, 1982.

[30] N. Tishby, "A dynamical systems approach to speech processing," in *Proc. Int. Conf. Acoust. Speech Sign. Process.*, (Albuquerque), pp. 365–368, 1990.

[31] S.-H. Wang, E. Paksov, and A. Gersho, "Performance of nonlinear prediction of speech," in *Proc. Int. Conf. Spoken Language Process.*, (Kobe), 1990.

[32] B. Townshend, "Nonlinear prediction of speech," in *Proc. Int. Conf. Acoust. Speech Sign. Process.*, (Toronto), pp. 425–428, 1991.

[33] J. Thyssen, H. Nielsen, and S. Hansen, "Nonlinear short-term prediction in speech coding," in *Proc. Int. Conf. Acoust. Speech Sign. Process.*, (Adelaide), pp. I185–I188, 1994.

[34] F. Tzeng, "Near optimal linear predictive speech coding," in *Proc. IEEE Global Telecomm. Conf.*, pp. 962–966, 1990.

[35] W. F. LeBlanc and V. Cuperman, "Sequential optimization of CELP for speech coding at 4kb/s," in *Abstracts IEEE Workshop on Speech Coding for Telecomm.*, (Whistler), pp. 105–106, 1991.

[36] S. Saito and K. Nakata, *Fundamentals of Speech Signal Processing.* New York: Academic Press, 1985.

[37] A. Papoulis, *Probability, Random Variables and Stochastic Processes, 2nd ed.* New York: McGraw-Hill, 1984.

[38] J. Makhoul, "Linear prediction: A tutorial review," *Proc. of the IEEE*, vol. 63, no. 4, pp. 561–580, 1975.

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

[40] M. L. Honig and D. G. Messerschmitt, *Adaptive Filters.* Dordrecht, Holland: Kluwer Academic Publishers, 1984.

[41] J. Schur, "Uber Potenzreihen, die in Innern des Einheitskreises beschrankt sind," *J. fuer die Reine and Angewandte Mathematiek*, vol. 147, pp. 205–232, 1917.

[42] J. Leroux and C. Gueguen, "A fixed point computation of partial correlation coefficients," *IEEE Trans. Acoust. Speech Signal Process.*, vol. ASSP-25, no. 3, pp. 257–259, 1979.

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

[44] J. L. Kelly and C. Lochbaum, "Speech synthesis," in *Proc. Stockholm Speech Communications Seminar*, (R.I.T., Stockholm), 1962.

[45] H. Wakita, "Direct estimation of the vocal tract shape by inverse filtering of the acoustic speech waveforms," *IEEE Trans. Audio and Electroacoustics*, vol. AU-21, no. 5, pp. 417–427, 1973.

[46] B. S. Atal and M. R. Schroeder, "Predictive coding of speech signals and subjective error criteria," *IEEE Trans. Speech Signal Proc.*, vol. ASSP-27, no. 3, pp. 247–254, 1979.

[47] W. B. Kleijn, R. P. Ramachandran, and P. Kroon, "Interpolation of the pitch-predictor parameters in analysis-by-synthesis speech coders," *IEEE Trans. Speech and Audio Process.*, vol. 2, no. 1, pp. 42–54, 1994.

[48] B. S. Atal and J. R. Remde, "A new model of LPC excitation for producing natural sounding speech at low bit rates," in *Proc. Int. Conf. Acoust. Speech Sign. Process.*, (Paris), pp. 614–617, 1982.

[49] E. F. Deprettere and P. Kroon, "Regular excitation reduction for effective and efficient LP-coding of speech," in *Proc. Int. Conf. Acoust. Speech Sign. Process.*, (Tampa), pp. 965–968, 1985.

[50] B. S. Atal and M. R. Schroeder, "Stochastic coding of speech at very low bit rates," in *Proc. Int. Conf. Comm.*, (Amsterdam), pp. 1610–1613, 1984.

[51] W. B. Kleijn, D. J. Krasinski, and R. H. Ketchum, "Analysis and improvement of the vector quantization in SELP," in *Signal Processing IV: Theories and Applications* (J. Lacoume, A. Chehikian, N. Martin, and J. Malbos, eds.), pp. 1043–1046, Amsterdam: Elsevier Science Publishers, 1988.

[52] S. Singhal and B. S. Atal, "Improving performance of multi-pulse LPC coders at low bit rates," in *Proc. Int. Conf. Acoust. Speech Sign. Process.*, (San Diego), pp. 1.3.1–1.3.4, 1984.

[53] P. Kroon, 1991. personal pommunication.

[54] J. H. Chen and A. Gersho, "Real-time vector APC speech coding at 4800 bps with adaptive postfiltering," in *Proc. Int. Conf. Acoust. Speech Sign. Process.*, (Dallas), pp. 2185–2188, 1987.

[55] R. Salami, C. Laflamme, J.-P. Adoul, and D. Massaloux, "A toll quality 8 kb/s speech codec for the personal communications system (pcs)," *IEEE Trans. Vehic. Techn.*, vol. 43, no. 3, pp. 808–816, 1994.

[56] J. H. Chen and A. Gersho, "Adaptive postfiltering for quality enhancement of coded speech," *IEEE Trans. Speech Audio Process.*, vol. 3, no. 1, 1995.

[57] F. Itakura, "Line spectrum representation of linear predictive coefficients," *J. Acoust. Soc. Am.*, vol. 57 Supplement, no. 1, p. S35, 1975.

[58] F. K. Soong and B.-H. Huang, "Line spectrum pair (LSP) and speech data compression," in *Proc. Int. Conf. Acoust. Speech Sign. Process.*, (San Diego), pp. 1.10.1–1.10.4, 1984.

[59] B. S. Atal, R. V. Cox, and P. Kroon, "Spectral quantization and interpolation for CELP

coders," in *Proc. Int. Conf. Acoust. Speech Sign. Process.*, (Glasgow), pp. 69–72, 1989.

[60] K. K. Paliwal and B. S. Atal, "Efficient vector quantization of LPC parameters at 24 bits/frame," *IEEE Trans. Speech Audio Process.*, vol. 1, no. 1, pp. 3–14, 1993.

[61] F. Soong, R. Cox, and N. Jayant, "On the backward adaptive predictor and optimal time-frequency bit assignments of a high performance subband coder," in *Proc. Int. Conf. Acoust. Speech Sign. Process.*, (Tokyo), pp. 1343–1346, 1986.

[62] B. S. Atal and S. L. Hanauer, "Speech analysis and synthesis by linear prediction of the speech wave," *J. Acoust. Soc. Amer.*, vol. 50, no. 2, pp. 637–655, 1971.

[63] W. Hess, *Pitch Determination of Speech Signals*. Berlin: Springer Verlag, 1983.

[64] A. V. McCree and T. P. Barnwell, "Improving the performance of a mixed-excitation LPC vocoder in acoustic noise," in *Proc. Int. Conf. Acoust. Speech Sign. Process.*, (San Francisco), pp. II137–II138, 1992.

[65] L. B. Almeida and J. M. Tribolet, "Nonstationary spectral modeling of voiced speech," *IEEE Trans. Acoust. Speech, Signal Process.*, vol. ASSP-31, no. 3, pp. 664–677, 1983.

[66] R. J. McAulay and T. F. Quatieri, "Speech analysis/synthesis based on a sinusoidal representation," *IEEE Trans. Acoust. Speech Signal Process.*, vol. 34, pp. 744–754, 1986.

[67] D. W. Griffin and J. S. Lim, "Multiband excitation vocoder," *IEEE Trans. Acoust. Speech Sign. Process.*, vol. ASSP-36, no. 8, pp. 1223–1235, 1988.

[68] W. B. Kleijn, "Continuous representations in linear predictive coding," in *Proc. Int. Conf. Acoust. Speech Sign. Process.*, (Toronto), pp. 201–204, 1991.

[69] W. B. Kleijn, "On the periodicity of speech coded with linear-prediction based analysis-by-synthesis coders," *IEEE Trans. Speech and Audio Process.*, vol. 2, no. 4, pp. 539–542, 1994.

[70] M. A. Kohler and L. M. Supplee, "Progress towards a new government standard 2400 bps voice coder," in *Proc. Int. Conf. Acoust. Speech Sign. Process.*, (Detroit), pp. 488–451, IEEE, 1995.