14. Principles of Speech Coding

W. B. Kleijn

Speech coding is the art of reducing the bit rate required to describe a speech signal. In this chapter, we discuss the attributes of speech coders as well as the underlying principles that determine their behavior and their architecture. The ubiquitous class of linear-prediction-based coders is used as an illustration. Speech is generally modeled as a sequence of stationary signal segments, each having unique statistics. Segments are encoded using a two-step procedure: (1) find a model describing the speech segment, (2) encode the segment assuming it is generated by the model. We show that the bit allocation for the model (the predictor parameters) is independent of overall rate and of perception, which is consistent with existing experimental results. The modeling of perception is an important aspect of efficient coding and we discuss how various perceptual distortion measures can be integrated into speech coders.

14.1 The Objective of Speech Coding

In modern communication systems, speech is represented by a sequence of bits. The main advantage of this *binary* representation is that it can be recovered exactly (without distortion) from a noisy channel (assuming proper system design), and does not suffer from decreasing quality when transmitted over many transmission legs. In contrast, analog transmission generally results in an increase of distortion with the number of legs.

An acoustic speech signal is inherently analog. Generally, the resulting analog microphone output is converted to a binary representation in a manner consistent with Shannon's sampling theorem. That is, the analog signal is first band-limited using an anti-aliasing filter, and then simultaneously sampled and quantized. The output of the *analog-to-digital (A/D) converter* is a digital speech signal that consists of a sequence of numbers of finite precision, each representing a sample of the band-limited speech signal. Common sampling rates are 8 and 16 kHz, rendering *narrowband speech* and *wideband speech*, respectively, usually with a precision of 16 bits per sample. For the 8 kHz sampling rate a logarithmic 8-bit-per-sample representation is also common.

Particularly at the time of the introduction of the binary speech representation, the bit rate produced by the A/D converter was too high for practical applications such as cost-effective mobile communications and secure telephony. A search ensued for more-efficient digital representations. Such representations are possible since the digital speech contains *irrelevancy* (the signal is described with a higher precision than is needed) and *redundancy* (the rate can be decreased without affecting precision). The aim was to trade off computational effort at the transmitter and receiver for the bit rate required for the speech representation. Efficient representations generally involve a *model* and a set of *model parameters*, and sometimes a set of coefficients that form the input to the model. The algorithms used to reduce the required rate are called speech-coding algorithms, or *speech codecs*.

The performance of speech codecs can be measured by a set of properties. The fundamental codec attributes are bit rate, speech quality, quality degradation due to channel errors and packet loss, delay, and computational effort. Good performance for one of the attributes generally leads to lower performance for the others. The interplay between the attributes is governed by the fundamental laws of information theory, the properties of the speech signal, limitations in our knowledge, and limitations of the equipment used.

To design a codec, we must know the desired values for its attributes. A common approach to develop a speech codec is to constrain all attributes but one quantitatively. The design objective is then to optimize the remaining attribute (usually quality or rate) subject to these constraints. A common objective is to maximize the average quality over a given set of channel conditions, given the rate, the delay, and the computational effort.

In this chapter, we attempt to discuss speech coding at a generic level and yet provide information useful for practical coder design and analysis. Section 14.2 describes the basic attributes of a speech codec. Section 14.3 discusses the underlying principles of coding and Sect. 14.4 applies these principles to a commonly used family of linear predictive (autoregressive model-based) coders. Section 14.5 discusses distortion criteria and how they affect the architecture of codecs. Section 14.6 provides a summary of the chapter.

14.2 Speech Coder Attributes

The usefulness of a speech coder is determined by its attributes. In this section we describe the most important attributes and the context in which they are relevant in some more detail. The attributes were earlier discussed in [14.1, 2].

14.2.1 Rate

The rate of a speech codec is generally measured as the average number of bits per second. For *fixed-rate* coders the bit rate is the same for each coding block, while for *variable-rate* coders it varies over time.

In traditional circuit-switched communication systems, a fixed rate is available for each communication direction. It is then natural to exploit this rate at all times, which has resulted in a large number of standardized fixed-rate speech codecs. In such coders each particular parameter or variable is encoded with the same number of bits for each block. This a priori knowledge of the bit allocation has a significant effect on the structure of the codec. For example, the mapping of the quantization indices to the transmitted codewords is trivial. In more-flexible circuit-switched networks (e.g., modern mobile-phone networks), codecs may have a variable number of modes, each mode having a different fixed rate [14.3,4]. Such codecs with a set of fixed coding rates should not be confused with true variable-rate coders.

In variable-rate coders, the bit allocation within a particular block for the parameters or variable depends on the signal. The bit allocation for a parameter varies with the quantization index and the mapping from the quantization index to the transmitted codeword is performed by means of a table lookup or computation, which can be very complex. The major benefit of variable-rate coding is that it leads to higher coding efficiency than fixed-rate coders because the rate constraint is less strict.

In general, network design evolves towards the facilitation of variable-rate coders. In packet-switched communication systems, both packet rate and size can vary, which naturally leads to variable-rate codecs. While variable-rate codecs are common for audio and video signals, they are not yet commonplace for speech. The requirements of low rates and delays lead to a small packet payload for speech signals. The relatively large packet header size limits the benefits of the low rate and,

consequently, the benefit of variable-rate speech coding. However, with the removal of the fixed-rate constraint, it is likely that variable-rate speech codecs will become increasingly common.

14.2.2 Quality

To achieve a significant rate reduction, the parameters used to represent the speech signal are generally transmitted at a reduced precision and the reconstructed speech signal is not a perfect copy of the original digital signal. It is therefore important to ensure that its quality meets a certain standard.

In speech coding, we distinguish two applications for quality measures. First, we need to evaluate the *overall quality* of a particular codec. Second, we need a *distortion measure* to decide how to encode each signal block (typically of duration 5–25 ms). The distortion measure is also used during the design of the coder (in the training of its codebooks). Naturally, these quality measures are not unrelated, but in practice their formulation has taken separate paths. Whereas overall quality can be obtained directly from scoring of speech utterances by humans, distortion measures used in coding algorithms have been defined (usually in an ad hoc manner) based on knowledge about the human auditory system.

The only true measure of the overall quality of a speech signal is its rating by humans. Standardized conversational and listening tests have been developed to obtain reliable and repeatable (at least to a certain accuracy) results. For speech coding, listening tests, where a panel of listeners evaluates performance for a given set of utterances, are most common. Commonly used standardized listening tests use either an absolute category rating, where listeners are asked to score an utterance on an absolute scale, or a degradation category rating, where listeners are asked to provide a relative score. The most common overall measure associated with the absolute category rating of speech quality is the mean opinion score (MOS) [14.5]. The MOS is the mean value of a numerical score given to an utterance by a panel of listeners, using a standardized procedure. To reduce the associated cost, subjective measures can be approximated by objective, repeatable algorithms for many practical purposes. Such measures can be helpful in the development of new speech coders. We refer to [14.6–8] and to Chap. 5 for more detail on the subject of overall speech quality.

As a distortion measure for speech segments variants of the squared-error criterion are most commonly used. The squared-error criterion facilitates fast evaluation for coding purposes. Section 14.5 discusses distortion measures in more detail. It is shown that adaptively weighted squared error criteria can be used for a large range of perceptual models.

14.2.3 Robustness to Channel Imperfections

Early terrestrial digital communication networks were generally designed to have very low error rates, obviating the need for measures to correct errors for the transmission of speech. In contrast, bit errors and packet loss are inherent in modern communication infrastructures.

Bit errors are common in wireless networks and are generally addressed by introducing channel codes. While the integration of source and channel codes can result in higher performance, this is not commonly used because it results in reduced modularity. Separate source and channel coding is particularly advantageous when a codec is faced with different network environments; different channel codes can then be used for different network conditions.

In packet networks, the open systems interconnection reference (OSI) model [14.9] provides a separation of various communication functionalities into seven layers. A speech coder resides in the application layer, which is the seventh and highest layer. Imperfections in the transmission are removed in both the physical layer (the first layer) and the transport layer (the fourth layer). The physical layer removes *soft* information, which consists of a probability for the allowed symbols, and renders a sequence of bits to the higher layers. Error control normally resides in the transport layer. However, the error control of the transport layer, as specified by the transmission control protocol (TCP) [14.10], and particularly the automatic repeat requests that TCP uses is generally not appropriate for real-time communication of audiovisual data because of delay. TCP is also rarely used for broadcast and multicast applications to reduce the load on the transmitter. Instead, the user datagram protocol (UDP) [14.11] is used, which means that the coded signal is handed up to the higher network layers without error correction. It is possible that in future systems cross-layer interactions will allow the application layer to receive information about the soft information available at the physical layer.

Handing the received coded signal with its defects directly to the application layer allows the usage of both the inherent redundancy in the signal and our knowledge of the perception of distortion by the user. This leads to coding systems that exhibit a graceful degrada-

tion with increasing error rate. We refer to the chapter on voice over internet protocol (IP) for more detail on techniques that lead to robustness against bit errors and packet loss.

14.2.4 Delay

From coding theory [14.12], we know that optimal coding performance generally requires a delay in the transfer of the message. Long delays are impractical because they are generally associated with methods with high computational and storage requirements, and because in real-time environments (common for speech) the user does not tolerate a long delay.

Significant delay directly affects the quality of a conversation. Impairment to conversations is measurable at one-way delays as low as 100 ms [14.13], although 200 ms is often considered a useful bound.

Echo is perceivable at delays down to 20 ms [14.14]. Imperfections in the network often lead to so-called network echo. Low-delay codecs have been designed to keep the effect of such echo to a minimum, e.g., [14.15]. However, echo cancelation has become commonplace in communication networks. Moreover, packet networks have an inherent delay that requires echo cancelation even for low-delay speech codecs. Thus, for most applications codecs can be designed without consideration of echo.

In certain applications the user may hear both an acoustic signal and a signal transmitted by a network. Examples are flight control rooms and wireless systems for hearing-impaired persons. In this class of applications, coding delays of less than 10 ms are needed to attain an acceptable overall delay.

14.2.5 Computational and Memory Requirements

Economic cost is generally a function of the computational and memory requirements of the coding system. A common measure of computational complexity used in applications is the number of instructions required on a particular silicon device. This is often translated into the number of channels that can be implemented on a single device.

A complicating factor is that speech codecs are commonly implemented on fixed-point signal processing devices. Implementation on a fixed-point device generally takes significant development effort beyond that of the development of the floating-point algorithm.

It is well known that vector quantization facilitates an optimal rate versus quality trade-off. Basic vector quantization techniques require very high computational effort and the introduction of vector quantization in speech coding resulted in promising but impractical codecs [14.16]. Accordingly, significant effort was spent to develop vector quantization structures that facilitate low computational complexity [14.17–19]. The continuous improvement in vector quantization methods and an improved understanding of the advantages of vector quantization over scalar quantization [14.20, 21] has meant that the computational effort of speech codecs has not changed significantly over the past two decades, despite significant improvement in codec performance. More effective usage of scalar quantization and the development of effective lattice vector quantization techniques make it unlikely that the computational complexity of speech codecs will increase significantly in the future.

14.3 A Universal Coder for Speech

In this section, we consider the encoding of a speech signal from a fundamental viewpoint. In information-theoretic terminology, speech is our *source* signal. We start with a discussion of the direct encoding of speech segments, without imposing any structure on the coder. This discussion is not meant to lead directly to a practical coding method (the computational effort would not be reasonable), but to provide an insight into the structure of existing coders. We then show how a signal model can be introduced. The signal model facilitates coding at a reasonable computational cost and the resulting coding paradigm is used by most speech codecs.

14.3.1 Speech Segment as Random Vector

Speech coders generally operate on a sequence of subsequent signal segments, which we refer to as *blocks* (also commonly known as *frames*). Blocks consist generally, but not always, of a fixed number of samples. In the present description of a basic coding system, we divide the speech signal into subsequent blocks of equal length and denote the block length in samples by k. We neglect dependencies across block boundaries, which is not always justified in a practical implementation, but simplifies the discussion; it is generally straightforward to cor-

rect this omission on implementation. We assume that the blocks can be described by k-dimensional random vectors $\boldsymbol{X}^k$ with a probability density function $p_{\boldsymbol{X}^k}(\boldsymbol{x}^k)$ for any $\boldsymbol{x}^k \in \mathbb{R}^k$, the k-dimensional Euclidian space (following convention, we denote random variables by capital letters and realizations by lower case letters).

For the first part of our discussion (Sect. 14.3.2), it is sufficient to assume the existence of the probability density function. It is natural, however, to consider some structure of the probability density $p_{\boldsymbol{X}^k}(\cdot)$ based on the properties of speech. We commonly describe speech in terms of a particular set of sounds (a distinct set of phones). A speech vector then corresponds to one sound from a countable set of speech sounds. We impose the notion that speech consists of a set of sounds on our probabilistic speech description. We can think of each sound as having a particular probability density. A particular speech vector then has one of a set of possible probability densities. Each member probability density of the set has an a *prior* probability, denoted as $p_I(i)$, where i indexes the set. The prior probability $p_I(i)$ is the probability that a random vector $\boldsymbol{X}^k$ is drawn from the particular member probability density i. The overall probability function of the random speech vector $p_{\boldsymbol{X}^k}(\cdot)$ is then a *mixture* of probability density functions

$$p_{\boldsymbol{X}^k}(\boldsymbol{x}^k) = \sum_{i \in \mathcal{A}} p_I(i) p_{\boldsymbol{X}^k|I}(\boldsymbol{x}^k|i) \,, \tag{14.1}$$

where $\mathcal{A}$ is the set of indexes for component densities and $p_{\boldsymbol{X}^k|I}(\cdot|\cdot)$ is the density of component i. These densities are commonly referred to as *mixture components*. If the set of mixture components is characterized by continuous parameters, then the summation must be replaced by an integral.

A common motivation for the mixture formulation of (14.1) is that a good approximation to the true probability density function can be achieved with a mixture of a finite set of probability densities from a particular family. This eliminates the need for the physical motivation. The family is usually derived from a single *kernel* function, such as a Gaussian. The kernel is selected for mathematical tractability.

If a mixture component does correspond to a physically reasonable speech sound, then it can be considered a statistical *model* of the signal. As described in Sect. 14.3.4, it is possible to interpret existing speech coding paradigms from this viewpoint. For example, linear prediction identifies a particular autoregressive model appropriate for a block. Each of the autoregressive models of speech has a certain prior probability and this in turn leads to an overall probability for the speech vector. According to this interpretation, mixture models have long been standard tools in speech coding, even if this was not explicitly stated.

The present formalism does not impose stationarity conditions on the signal within the block. In the mixture density, it is reasonable to include densities that correspond to signal transitions. In practice, this is not common, and the probability density functions are usually defined based on the definition that the signal is stationary within a block. On the other hand, the assumption that all speech blocks are drawn from the same distribution is implicit in the commonly used coding methods. It is consistent with our neglect of interblock dependencies. Thus, if we consider the speech signal to be a vector signal, then we assume stationarity for this vector signal (which is a rather inaccurate approximation). Strictly speaking, we do not assume ergodicity, as averaging over a database is best interpreted an averaging over an ensemble of signals, rather than time averaging over a single signal.

14.3.2 Encoding Random Speech Vectors

To encode observed speech vectors $\boldsymbol{x}^k$ that form *realizations* of the random vector $\boldsymbol{X}^k$, we use a speech codebook $C_{\boldsymbol{X}^k}$ that consists of a countable set of k-dimensional vectors (the code vectors). We can write $C_{\boldsymbol{X}^k} = \{\boldsymbol{c}_q^k\}_{q \in \mathcal{Q}}$, where $\boldsymbol{c}_q^k \in \mathbb{R}^k$ and $\mathcal{Q}$ is a countable (but not necessarily finite) set of indices. A decoded vector is simply the entry of the codebook that is pointed to by a transmitted index.

The encoding with codebook vectors results in the removal of both redundancy and irrelevancy. It removes irrelevancy by introducing a reduced precision version of the vector $\boldsymbol{x}^k$, i. e., by quantizing $\boldsymbol{x}^k$. The quantized vector requires fewer bits to encode than the unquantized vector. The mechanism of the redundancy removal depends on the coding method and will be discussed in Sect. 14.3.3.

We consider the speech vector, $\boldsymbol{X}^k$, to have a continuous probability density function in $\mathbb{R}^k$. Thus, coding based on the finite-size speech codebook $C_{\boldsymbol{X}^k}$ introduces *distortion*. To minimize the distortion associated with the coding, the encoder selects the code vector (codebook entry) $\boldsymbol{c}_q^k$ that is nearest to the observed vector $\boldsymbol{x}^k$ according to a particular distortion measure,

$$q = \operatorname{argmin}_{q' \in \mathcal{Q}} d\left(\boldsymbol{x}^k, \boldsymbol{c}_{q'}^k\right) . \tag{14.2}$$

Quantization is the operation of finding the nearest neighbor in the codebook. The set of speech vectors

that is mapped to a particular code vector $\boldsymbol{c}_q^k$ is called a *quantization cell* or *Voronoi region*. We denote the Voronoi region as $\mathcal{V}_q$,

$$\mathcal{V}_q = \left\{\boldsymbol{x}^k : d\left(\boldsymbol{x}^k, \boldsymbol{c}_q^k\right) < d\left(\boldsymbol{x}^k, \boldsymbol{c}_m^k\right) \forall_{m \neq q}\right\} , \qquad (14.3)$$

where we have ignored that generally points exist for which the inequality is not strict. These are boundary points that can be assigned to any of the cells that share the boundary.

Naturally, the average [averaged over $p_{\boldsymbol{X}^k}(\cdot)$] distortion of the decoded speech vectors differs for different codebooks. A method for designing a coder is to find the codebook, i. e., the set $C_{\boldsymbol{X}^k} = \{\boldsymbol{c}_q^k\}_{q\in\mathcal{Q}}$, that minimizes the average distortion over the speech probability density, given a constraint on the transmission rate. It is not known how to solve this problem in a general manner. Iterative methods (the Lloyd algorithm and its variants, e.g., [14.22–24]) have been developed for the case where $|\mathcal{Q}|$ (the *cardinality* or number of vectors in $C_{\boldsymbol{X}^k}$) is finite. The iterative approach is not appropriate for our present discussion for two reasons. First, we ultimately are interested in structured quantizers that allow us to approximate the optimal codebook and structure is difficult to determine from the iterative method. Second, as we will see below for the constrained-entropy case, practical codebooks do not necessarily have finite cardinality. Instead of the iterative approach, we use an approach where we make simplifying assumptions, which are asymptotically accurate for high coding rates.

14.3.3 A Model of Quantization

To analyze the behavior of the speech codebook, we construct a model of the quantization (encoding–decoding) operation. (This *quantization* model is not to be confused with the probabilistic *signal* model described in the next subsection.) Thus, we make the quantization problem mathematically tractable. For simplicity, we use the squared error criterion (Sect. 14.5 shows that this criterion can be used over a wide range of coding scenarios). We also make the standard assumption that the quantization cells are convex (for any two points in a cell, all points on the line segment connecting the two points are in the cell). To construct our encoding–decoding model, we make three additional assumptions that cannot always be justified:

1. The density $p_{\boldsymbol{X}^k}(\boldsymbol{x}^k)$ is constant within each quantization cell. This implies that the probability that a speech vector is inside a cell with index q is

$$p_Q(q) = V_q p_{\boldsymbol{X}^k}(\boldsymbol{x}^k), \boldsymbol{x}^k \in \mathcal{V}_q, \qquad (14.4)$$

where V_q is the volume of the k-dimensional cell.

2. The average distortion for speech data falling within cell q is

$$D_q = C V_q^{\frac{2}{k}} , \qquad (14.5)$$

where C is a constant. The assumption made in (14.5) essentially means that the cell shape is fixed. *Gersho* [14.25], conjectured that this assumption is correct for optimal codebooks.

3. We assume that the countable set of code vectors $C_{\boldsymbol{X}^k}$ can be represented by a code-vector density, denoted as $g(\boldsymbol{x}^k)$. This means that the cell volume now becomes a function of $\boldsymbol{x}^k$ rather than the cell index q; we replace V_q by $V(\boldsymbol{x}^k)$. To be consistent we must equate the density with the inverse of the cell volume:

$$g(\boldsymbol{x}^k) = \frac{1}{V(\boldsymbol{x}^k)} . \qquad (14.6)$$

The third assumption also implies that we can replace D_q by $D(\boldsymbol{x}^k)$.

The three assumptions listed above lead to solutions that can generally be shown to hold asymptotically in the limit of infinite rate. The theory has been observed to make reasonable predictions of performance for practical quantizers at rates down to two bits per dimension [14.26, 27], but we do not claim accuracy here. The theory serves as a vehicle to understand quantizer behavior and *not* as an accurate predictor of performance.

The code-vector density $g(\boldsymbol{x}^k)$ of our quantization model replaces the set of code vectors as the description of the codebook. Our objective of finding the codebook that minimizes the average distortion subject to a rate constraint has become the objective of finding the optimal density $g(\boldsymbol{x}^k)$ that minimizes the distortion

$$\begin{aligned} D &= \int D(\boldsymbol{x}^k) p_{\boldsymbol{X}^k}(\boldsymbol{x}^k) \mathrm{d}\boldsymbol{x}^k \\ &= C \int V(\boldsymbol{x}^k)^{\frac{2}{k}} p_{\boldsymbol{X}^k}(\boldsymbol{x}^k) \mathrm{d}\boldsymbol{x}^k \\ &= C \int g(\boldsymbol{x}^k)^{-\frac{2}{k}} p_{\boldsymbol{X}^k}(\boldsymbol{x}^k) \mathrm{d}\boldsymbol{x}^k , \end{aligned} \qquad (14.7)$$

subject to a rate constraint.

Armed with our quantization model, we now attempt to find the optimal density $g(\boldsymbol{x}^k)$ (the optimal codebook) for encoding speech. We consider separately two commonly used constraints on the rate: a given fixed rate and a given average rate. As mentioned in Sect. 14.2.1, the former rate constraint applies to circuit-switched networks and the latter rate constraint represents situations

where the rate can be varied continuously, such as, for example, in storage applications and packet networks.

We start with the fixed-rate requirement, where each codebook vector $\boldsymbol{c}_q^k$ is encoded with a codeword of a fixed number of bits. This is called *constrained-resolution* coding. If we use a rate of R bits per speech vector then we have a codebook cardinality of $N = 2^R$ and the density $g(\boldsymbol{x}^k)$ must be consistent with this cardinality:

$$N = \int_{\mathbb{R}^k} g(\boldsymbol{x}^k)\,\mathrm{d}\boldsymbol{x}^k \,. \tag{14.8}$$

We have to minimize the average distortion of (14.7) subject to the constraint (14.8) (i. e., subject to given N). This constrained optimization problem is readily solved with the calculus of variations. The solution is

$$\begin{aligned} g(\boldsymbol{x}^k) &= N \frac{p_{\boldsymbol{X}^k}(\boldsymbol{x}^k)^{\frac{k}{k+2}}}{\int p_{\boldsymbol{X}^k}(\boldsymbol{x}^k)^{\frac{k}{k+2}}\,\mathrm{d}\boldsymbol{x}^k} \\ &= 2^R \frac{p_{\boldsymbol{X}^k}(\boldsymbol{x}^k)^{\frac{k}{k+2}}}{\int p_{\boldsymbol{X}^k}(\boldsymbol{x}^k)^{\frac{k}{k+2}}\,\mathrm{d}\boldsymbol{x}^k} \\ &= 2^R \underline{p_{\boldsymbol{X}^k}(\boldsymbol{x}^k)^{\frac{k}{k+2}}} \,, \end{aligned} \tag{14.9}$$

where the underlining denotes normalization to unit integral over $\mathbb{R}^k$ and where R is the bit rate per speech vector. Thus, our encoding–decoding model suggests that, for constrained-resolution coding, the density of the code vectors varies with the data density. At dimensionalities $k >> 1$ the density of the code vectors approximates a simple scaling of the probability density of the speech vectors since $k/(k+2) \to 1$ with increasing k.

In the constrained-resolution case, *redundancy* is removed by placing the codebook vectors such that they reflect the density of the data vectors. For example, as shown by (14.9), regions of $\mathbb{R}^k$ without data have no vectors placed in them. This means no codewords are used for regions that have no data. If we had placed codebook vectors there, these would have been redundant. Note that scalar quantization of the k-dimensional random vector $\boldsymbol{X}^k$ would do precisely that. Similarly, regions of low data density get relatively few code vectors, reducing the number of codewords spent in such regions.

Next, we apply our quantization model to the case where the average rate is constrained. That is, the codeword length used to encode the cell indices q varies. Let us denote the random index associated with the random vector $\boldsymbol{X}^k$ as Q. The source coding theorem [14.12] tells us the lowest possible average rate for uniquely (so it can be decoded) encoding the indices with separate codewords is within one bit of the index entropy (in bits)

$$H(Q) = -\sum_{q\in\mathcal{Q}} p_Q(q)\log_2[p_Q(q)] \,. \tag{14.10}$$

The entropy can be interpreted as the average of a bit allocation, $-\log_2[p_Q(q)]$, for each index q. Neglecting the aforementioned *within one bit*, the average rate constraint is $H(Q) = R$, where R is the selected rate. For this reason, this coding method is known as *constrained-entropy* coding. This neglect is reasonable as the difference can be made arbitrarily small by encoding sequences of indices, as in arithmetic coding [14.28], rather than single indices. We minimize the distortion of (14.7) subject to the constraint (14.10), i. e., subject to given $H(Q) = R$. Again, the constrained optimization problem is readily solved with the calculus of variations. In this case the solution is

$$g(\boldsymbol{x}^k) = 2^{H(Q)-h(\boldsymbol{X}^k)} = 2^{R-h(\boldsymbol{X}^k)} \,, \tag{14.11}$$

where $h(\boldsymbol{X}^k) = -\int p_{\boldsymbol{X}^k}(\boldsymbol{x}^k)\log_2[p_{\boldsymbol{X}^k}(\boldsymbol{x}^k)]\,\mathrm{d}\boldsymbol{x}^k$ is the *differential entropy* of $\boldsymbol{X}^k$ in bits, and where $H(Q)$ is specified in bits. It is important to realize that special care must be taken if $p_{\boldsymbol{X}^k}(\cdot)$ is singular, i. e., if the data lie on a manifold.

Equation (14.11) implies that the the code vector density is uniform across $\mathbb{R}^k$. The number of code vectors is countably infinite despite the fact that the rate itself is finite. The codeword length $-\log_2[p_Q(q)]$ increases very slowly with decreasing probability $p_Q(q)$ and, roughly speaking, long codewords make no contribution to the mean rate.

In the constrained-entropy case, redundancy is removed through the lossless encoding of the indices. Given the probabilities of the code vectors, (ideal) lossless coding provides the most efficient bit assignment that allows unique decoding, and this rate is precisely the entropy of the indices. Code vectors in regions of high probability density receive short codewords and code vectors in regions of low probability density receive long codewords.

An important result that we have found for both the constrained-resolution and constrained-entropy cases is that the structure of the codebook is independent of the overall rate. The code-vector density simply increases as 2^R (cf. (14.9) and (14.11), respectively) anywhere in $\mathbb{R}^k$. Furthermore, for the constrained-entropy case, the code vector density depends only through the global variable $h(\boldsymbol{X}^k)$ on the probability density.

14.3.4 Coding Speech with a Model Family

Although the quantization model of Sect. 14.3.3 provides interesting results, a general implementation of

a codebook for the random speech vector $\boldsymbol{X}^k$ leads to practical problems, except for small k. For the constrained-resolution case, larger values of k lead to codebook sizes that do not allow for practical training procedures for storage on conventional media. For the constrained-entropy case, the codebook itself need not be stored, but we need access to the probability density of the codebook entries to determine the corresponding codewords (either offline or through computation during encoding). We can resolve these practical coding problems by using a *model* of the density. Importantly, to simplify the computational effort, we do *not* assume that the model is an accurate representation of the density of the speech signal vector, we simply make a best effort given the tools we have.

The model-based approach towards reducing computational complexity is suggested by the mixture model that we discussed in Sect. 14.3.1. If we classify each speech vector first as corresponding to a particular sound, then we can specify a probability density for that sound. A signal model specifies the probability density, typically by means of a formula for the probability density. The probability densities of the models are typically selected to be relatively simple. The signal models reduce computational complexity, either because they reduce codebook size or because the structural simplicity of the model simplifies the lossless coder. We consider models of a similar structure to be member of a *model family*. The selection of a particular model from the family is made by specifying *model parameters*.

Statistical signal models are commonly used in speech coding, with autoregressive modeling (generally referred to as linear prediction coding methods) perhaps being the most common. In this section, we discuss the selection of a particular model from the model family (i.e., the selection of the model parameters) and the balance in bit allocation between the model and the specification of the speech vector.

Our starting point is that a family of signal models is available for the coding operation. The model family can be any model family that provides a probability assignment for the speech vector $\boldsymbol{x}^k$. We discuss relevant properties for coding with signal models. We do not make the assumption that the resulting coding method is close to a theoretical performance bound on the rate versus distortion trade-off. As seid, we also do not make an assumption about the appropriateness of the signal model family for the speech signal. The model probabilities may not be accurate. However it is likely that models that are based on knowledge of speech production result in better performance.

The reasoning below is based on the early descriptions of the minimum description length (MDL) principle for finding signal models [14.29–31]. These methods separate a code for the model and a code for the signal realization, making them relevant to practical speech coding methods whereas later MDL methods use a single code. Differences from the MDL work include a stronger focus on distortion, and the consideration of the constrained-resolution case, which is of no interest to modeling theory.

Constrained-Entropy Case

First we consider the constrained-entropy case, i. e., we consider the case of a uniform codebook. Each speech vector is encoded with a codebook where each cell is of identical volume, which we denote as V. Let the model distribution be specified by a set of model parameters, θ. We consider the models to have a probability density, which means that a particular parameter set θ corresponds to a particular realization of a random parameter vector Θ. We write the probability density of $\boldsymbol{X}^k$ assuming the particular parameter set θ as $p_{\boldsymbol{X}^k;\Theta}(\boldsymbol{x}^k|\theta)$. The corresponding overall model density is

$$\tilde{p}_{\boldsymbol{X}^k}(\boldsymbol{x}^k) = -\sum_{\theta} p_{\boldsymbol{X}^k|\Theta}(\boldsymbol{x}^k|\theta) \cdot p_{\Theta}(\theta)\,, \qquad (14.12)$$

where the summation is over all parameter sets. The advantage of selecting and then using models $p_{\boldsymbol{X}^k|\Theta}(\boldsymbol{x}^k|\theta)$ from the family over using the composite model density $\tilde{p}_{\boldsymbol{X}^k}(\boldsymbol{x}^k)$ is a decrease of the computational effort.

The quantization model of Sect. 14.3.3 and in particular (14.4) and (14.10), show that the constrained-entropy encoding of a vector $\boldsymbol{x}^k$ assuming the model with parameters θ requires $-\log_2[Vp_{\boldsymbol{X}^k|\Theta}(\boldsymbol{x}^k|\theta)]$ bits. In addition, the decoder must receive side information specifying the model.

Let $\hat{\theta}(\boldsymbol{x}^k)$ be the parameter vector that maximizes $p_{\boldsymbol{X}^k|\Theta}(\boldsymbol{x}^k|\theta)$ and, thus, minimizes the bit allocation $-\log_2[Vp_{\boldsymbol{X}^k|\Theta}(\boldsymbol{x}^k|\theta)]$. That is, $p_{\boldsymbol{X}^k|\Theta}[\boldsymbol{x}^k|\hat{\theta}(\boldsymbol{x}^k)]$ is the maximum-likelihood model (from the family) for encoding the speech vector $\boldsymbol{x}^k$. The random speech vectors $\boldsymbol{X}^k$ do not form a countable set and as a result the random parameter vector $\hat{\Theta}(\boldsymbol{X}^k)$ generally does not form a countable set for conventional model families such as autoregressive models. To encode the model, we must discretize it.

To facilitate transmission of the random model index, J, the model parameters must be quantized and we write the random parameter set corresponding to random index J as $\theta(J)$. If $p_J(j)$ is a prior probability of the model index, the overall bit allocation for the vector

$\boldsymbol{x}^k$ when encoded with model j is

$$\begin{aligned} l &= -\log_2[p_J(j)] - \log_2\left\{V(\boldsymbol{x}^k)\, p_{\boldsymbol{X}^k|\Theta}[\boldsymbol{x}^k|\theta(j)]\right\} \\ &= -\log_2[p_J(j)] + \log_2\left(\frac{p_{\boldsymbol{X}^k|\Theta}[\boldsymbol{x}^k|\hat{\theta}(\boldsymbol{x}^k)]}{p_{\boldsymbol{X}^k|\Theta}[\boldsymbol{x}^k|\theta(j)]}\right) \\ &\quad - \log_2[V(\boldsymbol{x}^k)\, p_{\boldsymbol{X}^k|\Theta}(\boldsymbol{x}^k|\hat{\theta})]\,, \end{aligned} \tag{14.13}$$

where the term $\log_2\left(\frac{p_{\boldsymbol{X}^k|\Theta}[\boldsymbol{x}^k|\hat{\theta}(\boldsymbol{x}^k)]}{p_{\boldsymbol{X}^k|\Theta}[\boldsymbol{x}^k|\theta(j)]}\right)$ represents the additional (excess) bit allocation required to encode $\boldsymbol{x}^k$ with model j over the bit allocation required to encode $\boldsymbol{x}^k$ with the true maximum-likelihood model from the model family.

With some abuse of notation, we denote by $j(\boldsymbol{x}^k)$ the function that provides the index for a given speech vector $\boldsymbol{x}^k$. In the following, we assume that the functions $\theta(j)$ and $j(\boldsymbol{x}^k)$ minimize l. That is, we quantize θ so as to minimize the total number of bits required to encode $\boldsymbol{x}^k$.

We are interested in the bit allocation that results from averaging over the probability density $p_{\boldsymbol{X}^k}(\cdot)$ of the speech vectors,

$$\begin{aligned} \mathrm{E}\{\mathsf{L}\} = &-\mathrm{E}\left\{\log_2[p_J(j(\boldsymbol{X}^k))]\right\} \\ &-\mathrm{E}\left\{\log_2\left(\frac{p_{\boldsymbol{X}^k|\Theta}(\boldsymbol{X}^k|\theta(j(\boldsymbol{X}^k)))}{p_{\boldsymbol{X}^k|\Theta}(\boldsymbol{X}^k|\hat{\theta}(\boldsymbol{X}^k))}\right)\right\} \\ &-\mathrm{E}\left\{\log_2\left[V(\boldsymbol{X}^k)\, p_{\boldsymbol{X}^k|\Theta}(\boldsymbol{X}^k|\hat{\theta}(\boldsymbol{X}^k))\right]\right\}, \end{aligned} \tag{14.14}$$

where $\mathrm{E}\{\cdot\}$ indicates averaging over the speech vector probability density and where L is the random bit allocation that has I as realization. In (14.14), the first term describes the mean bit allocation to specify the model, the second term specifies the mean excess in bits required to encode $\boldsymbol{X}^k$ assuming $\theta(j(\boldsymbol{x}^k))$ instead of assuming the optimal $\hat{\theta}(\boldsymbol{x}^k)$, and the third term specifies the mean number of bits required to encode $\boldsymbol{X}^k$ if the optimal model is available. Importantly, only the third term contains the cell volume that determines the mean distortion of the speech vectors through (14.7).

Assuming validity of the encoding model of Sect. 14.3.3, the optimal trade-off between the bit allocation for the model index and the bit allocation for the speech vectors $\boldsymbol{X}^k$ depends only on the mean of

$$\begin{aligned} \eta = &-\log_2\{p_J[j(\boldsymbol{x}^k)]\} \\ &-\log_2\left(\frac{p_{\boldsymbol{X}^k|\Theta}[\boldsymbol{x}^k|\theta(j(\boldsymbol{x}^k))]}{p_{\boldsymbol{X}^k|\Theta}[\boldsymbol{x}^k|\hat{\theta}(\boldsymbol{x}^k)]}\right), \end{aligned} \tag{14.15}$$

which is referred to as the [14.32]. The goal is to find the functions $\theta(\cdot)$ and $j(\cdot)$ that minimize the index of resolvability over the ensemble of speech vectors. An important consequence of our logic is that these functions, and therefore the rate allocation for the model index, are dependent only on the excess rate and the probability of the quantized model. As the third term of (14.14) is missing, no relation to the speech distortion exists. That is *the rate allocation for the model index J is independent of distortion and overall bit rate*. While the theory is based on assumptions that are accurate only for high bit rates, this suggests that the bit allocation for the parameters becomes proportionally more important at low rates.

The fixed entropy for the model index indicates, for example, that for the commonly used linear-prediction-based speech coders, the rate allocation for the linear prediction parameters is independent of the overall rate of the coder. As constrained-entropy coding is not commonly used for predictive coding, this result is not immediately applicable to conventional speech coders. However, the new result we derive below is applicable to such coders.

Constrained-Resolution Case

Most current speech coders were designed with a constrained-resolution (fixed-rate) constraint, making it useful to study modeling in this context. We need some preliminary results. For a given model, with parameter set θ, and optimal code vector density, the average distortion over a quantization cell centered at location $\boldsymbol{x}^k$ can be written

$$\begin{aligned} D(\boldsymbol{x}^k) &= CV(\boldsymbol{x}^k)^{\frac{2}{k}} \\ &= Cg(\boldsymbol{x}^k)^{-\frac{2}{k}} \\ &= CN^{-\frac{2}{k}}\left[p_{\boldsymbol{X}^k|\Theta}(\boldsymbol{x}^k|\theta)^{\frac{k}{k+2}}\right]^{-\frac{2}{k}}, \end{aligned} \tag{14.16}$$

where we have used (14.7) and (14.9). We take the expectation of (14.16) with respect to the true probability density function $p_{\boldsymbol{X}^k}(\boldsymbol{x}^k)$ and obtain the mean distortion for the constrained-resolution case:

$$D_{\mathrm{CR}} = CN^{-\frac{2}{k}}\mathrm{E}\left\{\left[p_{\boldsymbol{X}^k|\Theta}(\boldsymbol{X}^k|\theta)^{\frac{k}{k+2}}\right]^{-\frac{2}{k}}\right\}. \tag{14.17}$$

Equation (14.17) can be rewritten as

$$\begin{aligned} \frac{2}{k}\log_2(N) = &\log_2\left(\mathrm{E}\left\{\left[p_{\boldsymbol{X}^k|\Theta}(\boldsymbol{X}^k|\theta)^{\frac{k}{k+2}}\right]^{-\frac{2}{k}}\right\}\right) \\ &-\log_2\left(\frac{D_{\mathrm{CR}}}{C}\right). \end{aligned} \tag{14.18}$$

We assume that k is sufficiently large that, in the region where $p_{\boldsymbol{X}^k}(\boldsymbol{x}^k)$ is significant, we can use the expansion

$u \approx 1 + \log(u)$ for the term $[p_{\boldsymbol{X}^k|\Theta}(\boldsymbol{x}^k|\theta)^{k/(k+2)}]^{-2/k}$ and write

$$\begin{aligned}
&\log\left(\mathrm{E}\left\{\left[p_{\boldsymbol{X}^k|\Theta}(\boldsymbol{X}^k|\theta)^{\frac{k}{k+2}}\right]^{-\frac{2}{k}}\right\}\right)\\
&\approx \log\left(1-\frac{2}{k}\mathrm{E}\left\{\log\left[p_{\boldsymbol{X}^k|\Theta}(\boldsymbol{X}^k|\theta)^{\frac{k}{k+2}}\right]\right\}\right)\\
&\approx -\frac{2}{k}\mathrm{E}\left\{\log\left[p_{\boldsymbol{X}^k|\Theta}(\boldsymbol{X}^k|\theta)^{\frac{k}{k+2}}\right]\right\} . \qquad (14.19)
\end{aligned}$$

Having completed the preliminaries, we now consider the encoding of a speech vector $\boldsymbol{x}^k$. Let $L_{(m)}$ be the fixed bit allocation for the model index. The total rate is then

$$\begin{aligned}
L &= L_{(m)} + L(\boldsymbol{x}^k)\\
&= L_{(m)} + \log_2(N)\\
&= L_{(m)} - \mathrm{E}\left\{\log_2\left[p_{\boldsymbol{X}^k|\Theta}(\boldsymbol{X}^k|\theta)^{\frac{k}{k+2}}\right]\right\}\\
&\quad - \frac{k}{2}\log_2\left(\frac{D_{\mathrm{CR}}}{C}\right) . \qquad (14.20)
\end{aligned}$$

The form of (14.20) shows that, given the assumptions made, we can define an *equivalent codeword length* $\log_2[p_{\boldsymbol{X}^k|\Theta}(\boldsymbol{X}^k|\theta)^{k/(k+2)}]$ for each speech codebook entry. The equivalent codeword length represents the spatial variation of the distortion. Note that this equivalent codeword length does *not* correspond to the true codeword length of the speech vector codebook, which is fixed for the constrained-resolution case. For a particular codebook vector $\boldsymbol{x}^k$, the equivalent codeword length is

$$\begin{aligned}
L &= L_{(m)} - \log_2\left[p_{\boldsymbol{X}^k|\Theta}(\boldsymbol{x}^k|\theta)^{\frac{k}{k+2}}\right]\\
&\quad - \frac{k}{2}\log_2\left(\frac{D_{\mathrm{CR}}}{C}\right) . \qquad (14.21)
\end{aligned}$$

Similarly to the constrained-entropy case, we can decompose (14.21) into a rate component that relates to the encoding of the model parameters, a component that describes the excess equivalent rate resulting from limiting the precision of the model parameters, and a rate component that relates to optimal encoding with optimal (uncoded) model parameters:

$$\begin{aligned}
L &= L_{(m)} - \log_2(p_{\boldsymbol{X}^k|\Theta}(\boldsymbol{x}^k|\theta(j))^{\frac{k}{k+2}}) - \frac{k}{2}\log_2\left(\frac{D_{\mathrm{CR}}}{C}\right)\\
&= L_{(m)} - \log_2\left(\frac{p_{\boldsymbol{X}^k|\Theta}(\boldsymbol{x}^k|\theta(j))^{\frac{k}{k+2}}}{p_{\boldsymbol{X}^k|\Theta}(\boldsymbol{x}^k|\hat{\theta}(\boldsymbol{x}^k))^{\frac{k}{k+2}}}\right)\\
&\quad - \log_2\left[p_{\boldsymbol{X}^k|\Theta}(\boldsymbol{x}^k|\hat{\theta}(\boldsymbol{x}^k))^{\frac{k}{k+2}}\right] - \frac{k}{2}\log_2\left(\frac{D_{\mathrm{CR}}}{C}\right) ,
\end{aligned}$$

(14.22)

We can identify the last two terms as the bit allocation for $\boldsymbol{x}^k$ for the optimal constrained-resolution model for the speech vector $\boldsymbol{x}^k$. The second term is the excess equivalent bit allocation required to encode the speech vector with model j over the bit allocation required for the optimal model from the model family. The first two terms determine the trade-off between the bits spent on the model, and the bits spent on the speech vectors. These two terms form the index of resolvability for the constrained-resolution case:

$$\eta = L_{(m)} - \log_2\left(\frac{p_{\boldsymbol{X}^k|\Theta}(\boldsymbol{x}^k|\theta(j))^{\frac{k}{k+2}}}{p_{\boldsymbol{X}^k|\Theta}(\boldsymbol{x}^k|\hat{\theta}(\boldsymbol{x}^k))^{\frac{k}{k+2}}}\right) . \qquad (14.23)$$

As for the constrained-entropy case, the optimal set of functions $\theta(j)$ and $j(\boldsymbol{x}^k)$ (and, therefore, the bit allocation for the model) are dependent only on the speech vector density for the constrained-resolution case. The rate for the model is independent of the distortion selected for the speech vector and of the overall rate. With increasing k, the second term in (14.23) and (14.15) becomes identical. That is the expression for the excess rate for using the quantized model parameters corresponding to model j instead of the optimal parameters is identical.

The independence of the model-parameter bit allocation of the overall codec rate for the constrained-entropy case is of great significance for practical coding systems. We emphasize again that this result is valid only under the assumptions made in Sect. 14.3.3. We expect the independence to break down at lower rates, where the codebook $C_{\boldsymbol{X}^k}$ describing the speech cannot be approximated by a density.

Table 14.1 Bit rates of the AMR-WB coder [14.4]

Rate (bits)	6.6	8.85	12.65	14.25	15.85	18.25	19.85	23.05
AR model	36	46	46	46	46	46	46	46
Pitch parameter	23	26	30	30	30	30	30	30
Excitation	48	80	144	176	208	256	288	352

The results described in this section are indeed supported, at least qualitatively, by the configuration of practical coders. Table 14.1 shows the most important bit allocations used in the adaptive-multirate wideband (AMR-WB) speech coder [14.4]. The AMR-WB coder is a constrained-resolution coder. It is seen that the design of the codec satisfies the predicted behavior: the bit allocation for the model parameters is essentially independent of the rate of the codec, except at low rates.

Model-Based Coding

In signal-model-based coding we assume the family is known to the encoder and decoder. An index to the specific model is transmitted. Each model corresponds to a unique speech-domain codebook. The advantage of the model-based approach is that the structure of the density is simplified (which is advantageous for constrained-entropy coding) and that the required number of codebook entries for the constrained-resolution case is smaller. This facilitates searching through the codebook and/or the definition of the lossless coder.

The main result of this section is that we can determine the set of codebooks for the models independently of the overall rate (and speech-vector distortion). The result is consistent with existing results. The result of this section leads to fast codec design as there is no need to check the best trade-off in bit allocation between model and signal quantization.

When encoding with a model-based coding it is advantageous first to identify the *best* model, encode the model index j, and then encode the signal using codebook $C_{\boldsymbol{X}^k,j}$ that is associated with that particular model j. The model selection can be made based on the index of resolvability.

14.4 Coding with Autoregressive Models

We now apply the methods of Sect. 14.3 to a practical model family. Autoregressive model families are commonly used in speech coding. In speech coding this class of coders is generally referred to as being based on *linear prediction*. We discuss coding based on a family that consists of a set of autoregressive models of a particular order (denoted as p). To match current practice, we consider the constrained-resolution case.

We first formulate the index of resolvability in terms of a spectral formulation of the autoregressive model. We show that this corresponds to the definition of a distortion measure for the model parameters. The distortion measure is approximated by the commonly used Itakura–Saito and log spectral distortion measures. Thus, starting from a squared error criterion for the speech signal, we obtain the commonly used (e.g., [14.33–37]) distortion measures for the linear-prediction parameters. Finally, we show that our reasoning leads to an estimate for the bit allocation for the model. We discuss how this result relates to results on autoregressive model estimation.

14.4.1 Spectral-Domain Index of Resolvability

Our objective is to encode a particular speech vector $\boldsymbol{x}^k$ using the autoregressive model. To facilitate insight, it is beneficial to make a spectral formulation of the problem. To this purpose, we assume that k is sufficiently large to neglect edge effects. Thus, we neglect the difference between circular and linear convolution.

The autoregressive model assumption implies that $\boldsymbol{x}^k$ has a multivariate Gaussian probability density

$$p_{\boldsymbol{X}^k|\Theta}(\boldsymbol{x}^k|\theta) = \frac{1}{\sqrt{2\pi\det(\mathbf{R}_\theta)}}\exp\left(-\frac{1}{2}\boldsymbol{x}^{kT}\mathbf{R}_\theta^{-1}\boldsymbol{x}^k\right) . \quad (14.24)$$

$\mathbf{R}_\theta$ is the *model* autocorrelation matrix

$$\mathbf{R}_\theta = \mathbf{A}^{-1}\mathbf{A}^{-\mathrm{H}} , \quad (14.25)$$

where $\mathbf{A}$ a lower-triangular Toeplitz matrix with first column $\sigma[1, a_1, a_2, \cdots, a_p, 0, \cdots, 0]^{\mathrm{T}}$, where the a_i are the autoregressive model parameters (linear-prediction parameters), and p is the autoregressive model order and the superscript H is the Hermitian transpose. Thus, the set of model parameters is $\theta = \{\sigma, a_1, \cdots, a_p\}_{i=1,\cdots,p}$. We note that typically $p = 10$ for 8 kHz sampling rate and $p = 16$ for 12 kHz and 16 kHz sampling rate.

When k is sufficiently large, we can perform our analysis in terms of power spectral densities. The transfer function of the autoregressive model is

$$A(z)^{-1} = \frac{\sigma}{1 + a_1 z^{-1} + \cdots + a_p z^{-p}} , \quad (14.26)$$

where σ is a gain. This corresponds to the model power spectral density

$$\mathbf{R}_\theta(z) = |A(z)|^{-2} \,. \tag{14.27}$$

In the following, we make the standard assumption that $A(z)$ is minimum-phase.

Next we approximate (14.24) in terms of power spectral densities and the transfer function of the autoregressive model. Using *Szegö*'s theorem [14.38], it is easy to show that, asymptotically in k,

$$\det(\mathbf{R}_\theta) = \exp\left\{\frac{k}{2\pi}\int_0^{2\pi}\log\left[\mathbf{R}_\theta(\mathrm{e}^{\mathrm{i}\omega})\right]\mathrm{d}\omega\right\} \,. \tag{14.28}$$

We also use the asymptotic equality

$$\begin{aligned}\frac{1}{2}\boldsymbol{x}^{kT}\mathbf{R}_\theta^{-1}\boldsymbol{x}^k &= \frac{1}{4\pi}\int_0^{2\pi}\frac{|x(\mathrm{e}^{\mathrm{i}\omega})|^2}{\mathbf{R}_\theta(\mathrm{e}^{\mathrm{i}\omega})}\mathrm{d}\omega \\ &= \frac{k}{4\pi}\int_0^{2\pi}\frac{R_x(\mathrm{e}^{\mathrm{i}\omega})}{\mathbf{R}_\theta(\mathrm{e}^{\mathrm{i}\omega})}\mathrm{d}\omega \,, \end{aligned} \tag{14.29}$$

where $x(z) = \sum_{i=0}^{k-1} x_i z^{-i}$ for $\boldsymbol{x}^k = (x_1, \cdots, x_k)$ and $R_x(\mathrm{e}^{\mathrm{i}\omega}) = \frac{1}{k}|x(\mathrm{e}^{\mathrm{i}\omega})|^2$.

Equations (14.28) and (14.29) can be used to rewrite the multivariate density of (14.24) in terms of power spectral densities. It is convenient to write the log density:

$$\begin{aligned}&\log[p_{\boldsymbol{X}^k|\Theta}(\boldsymbol{x}^k|\theta)] \\ &= -\frac{1}{2}\log(2\pi) - \frac{k}{4\pi}\int_0^{2\pi}\log(\mathbf{R}_\theta(\mathrm{e}^{\mathrm{i}\omega}))\mathrm{d}\omega \\ &\quad - \frac{k}{4\pi}\int_0^{2\pi}\frac{R_x(\mathrm{e}^{\mathrm{i}\omega})}{\mathbf{R}_\theta(\mathrm{e}^{\mathrm{i}\omega})}\mathrm{d}\omega \,. \end{aligned} \tag{14.30}$$

We use (14.30) to find the index of resolvability for the constrained-resolution case. We make the approximation that k is sufficiently large that it is reasonable to approximate the exponent $k/(k+2)$ by unity in (14.23). This implies that we do not have to consider the normalization in this equation. Inserting (14.30) into (14.23) results in

$$\begin{aligned}\eta = L_{(m)} + \frac{k}{4\pi}\int_0^{2\pi}\Bigg[&-\log\left(\frac{R_{\hat\theta}(\mathrm{e}^{\mathrm{i}\omega})}{\mathbf{R}_\theta(\mathrm{e}^{\mathrm{i}\omega})}\right) \\ &+\frac{R_x(\mathrm{e}^{\mathrm{i}\omega})}{\mathbf{R}_\theta(\mathrm{e}^{\mathrm{i}\omega})} - \frac{R_x(\mathrm{e}^{\mathrm{i}\omega})}{R_{\hat\theta}(\mathrm{e}^{\mathrm{i}\omega})}\Bigg]\mathrm{d}\omega \,. \end{aligned} \tag{14.31}$$

The maximum-likelihood estimate of the autoregressive model $\hat\theta$ given a data vector $\boldsymbol{x}^k$ is a well-understood problem, e.g., [14.39, 40]. The predictor parameter estimate of the standard Yule–Walker solution method has the same asymptotic density as the maximum-likelihood estimate [14.41].

To find the optimal bit allocation for the model we have to minimize the expectation of (14.31) over the ensemble of all speech vectors. We study the behavior of this minimization. For notational convenience we define a cost function

$$\begin{aligned}\psi(\theta,\hat\theta) = \frac{k}{4\pi}\int_0^{2\pi}\Bigg[&-\log\left(\frac{R_{\hat\theta}(\mathrm{e}^{\mathrm{i}\omega})}{\mathbf{R}_\theta(\mathrm{e}^{\mathrm{i}\omega})}\right) \\ &+\frac{R_x(\mathrm{e}^{\mathrm{i}\omega})}{\mathbf{R}_\theta(\mathrm{e}^{\mathrm{i}\omega})} - \frac{R_x(\mathrm{e}^{\mathrm{i}\omega})}{R_{\hat\theta}(\mathrm{e}^{\mathrm{i}\omega})}\Bigg]\mathrm{d}\omega \,. \end{aligned} \tag{14.32}$$

Let θ be a particular model from a countable model set $\mathcal{C}_\Theta(L_{(m)})$ with a bit allocation $L_{(m)}$ for the model. Finding the optimal model set $\mathcal{C}_\Theta(L_{(m)})$ is then equivalent to

$$\begin{aligned}&\min_{L_{(m)}\in\mathbb{N}} \mathrm{E}[\eta] \\ &= \min_{L_{(m)}\in\mathbb{N}}\left\{L_{(m)} + \min_{\mathcal{C}_\Theta(L_{(m)})}\mathrm{E}\Big[\min_{\theta\in\mathcal{C}_\Theta(L_{(m)})}\psi(\theta,\hat\theta)\Big]\right\} \,. \end{aligned} \tag{14.33}$$

If we write

$$D(L_{(m)}) = \min_{\mathcal{C}_\Theta(L_{(m)})}\mathrm{E}\Big[\min_{\theta\in\mathcal{C}_\Theta(L_{(m)})}\psi(\theta,\hat\theta)\Big] \tag{14.34}$$

then (14.33) becomes

$$\min_{L_{(m)}\in\mathbb{N}}\mathrm{E}[\eta] = \min_{L_{(m)}\in\mathbb{N}}\left[L_{(m)} + D(L_{(m)})\right] \,. \tag{14.35}$$

If we interpret $D(L_{(m)})$ as a minimum mean distortion, minimizing (14.35) is equivalent to finding a particular point on a rate-distortion curve. We can minimize the cost function of (14.34) for all $L_{(m)}$ and then select the $L_{(m)}$ that minimizes the overall expression of (14.35). Thus only one particular distortion level, corresponding to one particular rate, is relevant to our speech coding system. This distortion–rate pair for the model is dependent on the distribution of the speech models. Assuming that $D(L_{(m)})$ is once differentiable towards $L_{(m)}$, then (14.35) shows that its derivative should be -1 at the optimal rate for the model.

14.4.2 A Criterion for Model Selection

We started with the notion of using an autoregressive model family to quantize the speech signal. We found

that we could do so by first finding the maximum-likelihood estimate $\hat{\theta}$ of the autoregressive model parameters, then selecting from a set of models $\mathcal{C}_\Theta(L_{(m)})$ the model nearest to the maximum-likelihood model based on the cost function $\psi(\theta,\hat{\theta})$ and then quantizing the speech given the selected model. As quantization of the predictor parameters corresponds to our model selection, it is then relevant to compare the distortion measure of (14.32) with the distortion measures that are commonly used for the linear-prediction parameters in existing speech coders.

To provide insight, it is useful to write $R_x(\mathrm{e}^{\mathrm{i}\omega}) = R_{\hat{\theta}}(\mathrm{e}^{\mathrm{i}\omega})\, R_w(\mathrm{e}^{\mathrm{i}\omega})$, where $R_w(\mathrm{e}^{\mathrm{i}\omega})$ represents a *remainder* power-spectral density that captures the spectral error of the maximum likelihood model. If the model family is of low order, then $R_w(\mathrm{e}^{\mathrm{i}\omega})$ includes the spectral *fine structure*. We can rewrite (14.32) as

$$\psi(\theta,\hat{\theta}) = \frac{k}{4\pi}\int_0^{2\pi}\left[-\log\left(\frac{R_{\hat{\theta}}(\mathrm{e}^{\mathrm{i}\omega})}{\mathbf{R}_\theta(\mathrm{e}^{\mathrm{i}\omega})}\right) + \left(\frac{R_{\hat{\theta}}(\mathrm{e}^{\mathrm{i}\omega})}{\mathbf{R}_\theta(\mathrm{e}^{\mathrm{i}\omega})} - 1\right) R_w(\mathrm{e}^{\mathrm{i}\omega})\right] \mathrm{d}\omega\,. \quad (14.36)$$

Interestingly, (14.36) reduces to the well-known *Itakura–Saito criterion* [14.42] if $R_w(\mathrm{e}^{\mathrm{i}\omega})$ is set to unity.

It is common (e.g., [14.43]) to relate different criteria through the series expansion $u = 1+\log(u)+\frac{1}{2}[\log(u)]^2+\cdots$. Assuming small differences between the optimal model $\hat{\theta}$ and the model from the set θ, (14.36) can be written

$$\psi(\theta,\hat{\theta}) \cong \frac{k}{4\pi}\int_0^{2\pi}\left\{\left[R_w(\mathrm{e}^{\mathrm{i}\omega})-1\right]\log\left(\frac{R_{\hat{\theta}}(\mathrm{e}^{\mathrm{i}\omega})}{\mathbf{R}_\theta(\mathrm{e}^{\mathrm{i}\omega})}\right) + \frac{1}{2}R_w(\mathrm{e}^{\mathrm{i}\omega})\left[\log\left(\frac{R_{\hat{\theta}}(\mathrm{e}^{\mathrm{i}\omega})}{\mathbf{R}_\theta(\mathrm{e}^{\mathrm{i}\omega})}\right)\right]^2\right\} \mathrm{d}\omega\,. \quad (14.37)$$

Equation (14.37) needs to be accurate only for nearest neighbors of $\hat{\theta}$.

We can simplify (14.37) further. With our assumptions for the autoregressive models, $\mathbf{R}_\theta(z)$ is related to monic minimum-phase polynomials through (14.27) and the further assumption that their gains σ are identical (i. e., is not considered here), this implies that

$$\frac{1}{2\pi}\int_0^{2\pi}\log(\mathbf{R}_\theta)\,\mathrm{d}\omega = \frac{1}{2\pi}\int_0^{2\pi}\log(R_{\hat{\theta}})\,\mathrm{d}\omega = \log(\sigma^2)\,. \quad (14.38)$$

This means that we can rewrite (14.37) as

$$\psi(\theta,\hat{\theta}) \cong \frac{k}{4\pi}\int_0^{2\pi} R_w(\mathrm{e}^{\mathrm{i}\omega})\left\{\log\left(\frac{R_{\hat{\theta}}(\mathrm{e}^{\mathrm{i}\omega})}{\mathbf{R}_\theta(\mathrm{e}^{\mathrm{i}\omega})}\right) + \frac{1}{2}\left[\log\left(\frac{R_{\hat{\theta}}(\mathrm{e}^{\mathrm{i}\omega})}{\mathbf{R}_\theta(\mathrm{e}^{\mathrm{i}\omega})}\right)\right]^2\right\} \mathrm{d}\omega\,. \quad (14.39)$$

Equation (14.39) forms the basic measure that must be optimized for the selection of the model from a set of models, i. e., for the optimal quantization of the model parameters.

If we can neglect the impact of $R_w(z)$, then (using the result of (14.38)) minimizing (14.39) is equivalent to minimizing

$$\psi(\theta,\hat{\theta}) \cong \frac{k}{8\pi}\int_0^{2\pi}\left[\log\left(\frac{R_{\hat{\theta}}(\mathrm{e}^{\mathrm{i}\omega})}{\mathbf{R}_\theta(\mathrm{e}^{\mathrm{i}\omega})}\right)\right]^2 \mathrm{d}\omega\,, \quad (14.40)$$

which is the well-known *mean squared log spectral distortion*, scaled by the factor $k/4$. Except for this scaling factor, (14.39) is precisely the criterion that is commonly used (e.g., [14.35, 44, 45]) to evaluate performance of quantizers for autoregressive (AR) model parameters. This is not unreasonable as the neglected modeling error $R_w(\mathrm{e}^{\mathrm{i}\omega})$ is likely uncorrelated with the model quantization error.

14.4.3 Bit Allocation for the Model

The AR model is usually described with a small number of parameters (as mentioned, $p = 10$ is common for 8 kHz sampling rate). Thus, the spectral data must lie on a manifold of dimension p or less in the log spectrum space. At high bit allocations, where measurement noise dominates (see also the end of Sect. 14.3.3), the manifold dimension is p and the spectral distortion is expected to scale as

$$D(L_{(m)}) = \frac{k}{4}\beta^2 N_{\mathrm{AR}}^{-\frac{2}{p}} = \frac{k}{4}\beta^2\,\mathrm{e}^{-\frac{2}{p}L_{(m)}}\,, \quad (14.41)$$

where β is a constant and N_{AR} is the number of spectral models in the family and $L_{(m)} = \log(N_{\mathrm{AR}})$. At higher spectral distortion levels, it has been observed that the physics of the vocal tract constrains the dimensionality of the manifold. This means that (14.41) is replaced by

$$D(L_{(m)}) = \frac{k}{4}\beta^2\,\mathrm{e}^{-\frac{2}{\kappa}L_{(m)}} \quad (14.42)$$

with $\kappa < p$. This behavior was observed for trained codebooks over a large range in [14.44] (similar behavior was

observed for cepstral parameters in [14.46]) and for specific vowels in [14.47]. The results of [14.44] correspond to $\kappa = 7.1$ and $\beta = 0.80$.

The mean of (14.31) becomes

$$\mathrm{E}[\eta] = L_{(m)} + \frac{k}{4}\beta^2 \mathrm{e}^{-\frac{2}{\kappa}L_{(m)}} \,. \tag{14.43}$$

Differentiating towards $L_{(m)}$ we find that the optimal bit allocation for the AR model selection to be

$$L_{(m)} = \frac{\kappa}{2}\log\left(\beta^2\frac{k}{2\kappa}\right) \,, \tag{14.44}$$

which is logarithmically dependent on k. Using the observed data of [14.44], we obtain an optimal rate of about 17 bits for 8 kHz sampled speech at a 20 ms block size. The corresponding mean spectral distortion is about 1.3 dB. The distortion is similar to the mean estimation errors found in experiments on linear predictive methods on speech sounds [14.48].

The 17 bit requirement for the prediction parameter quantizer is similar to that obtained by the best available prediction parameter quantizers that operate on single blocks and bounds obtained for these methods [14.35, 49–52]. In these systems the lowest bit allocation for 20 ms blocks is about 20 bits. However, the performance of these coders is entirely based on the often quoted 1 dB threshold for transparency [14.35]. The definition of this empirical threshold is consistent with the conventional two-step approach: the model parameters are first quantized using a separately defined criterion, and the speech signal is quantized thereafter based on a weighted squared error criterion. In contrast, we have shown that a single distortion measure operating on the speech vector suffices for this purpose.

We conclude that the definition of a squared-error criterion for the speech signal leads to a bit allocation for the autoregressive model. No need exists to introduce perception based thresholds on log spectral distortion.

14.4.4 Remarks on Practical Coding

The two-stage approach is standard practice in linear-prediction-based (autoregressive-model-based) speech codecs. In the selection stage, weighted squared error criteria in the so-called line-spectral frequency (LSF) representation of the prediction parameters are commonly used, e.g., [14.34–37]. If the proper weighting is used, then the criterion can be made to match the log spectral distortion measure [14.53] that we derived above.

The second stage is the selection of a speech codebook entry from a codebook corresponding to the selected model. The separation into a set of models simplifies this selection. In general, this means that a speech-domain codebook must be available for each model. It was recently shown that the computational or storage requirements for optimal speech-domain codebooks can be made reasonable by using a single codebook for each set of speech sounds that are similar except for a unitary transform [14.54]. The method takes advantage of the fact that different speech sounds may have similar statistics after a suitable unitary transform and can, therefore, share a codebook. As the unitary transform does not affect the Euclidian distance, it also does not affect the optimality of the codebook.

In the majority of codecs the speech codebooks are generated in real time, with the help of the model obtained in the first stage. This approach is the so-called *analysis-by-synthesis* approach. It can be interpreted as a method that requires the *synthesis* of candidate speech vectors (our speech codebook), hence the name. Particularly common is the usage of the analysis-by-synthesis approach for the autoregressive model [14.16,55]. While the analysis-by-synthesis approach has proven its merit and is used in hundreds of millions of communication devices, it is not optimal. It was pointed out in [14.54] that analysis-by-synthesis coding inherently results in a speech-domain codebook with quantization cells that have a suboptimal shape, limiting performance.

14.5 Distortion Measures and Coding Architecture

An objective of coding is the removal of irrelevancy. This means that precision is lost and that we introduce a difference between the original and the decoded signal, the error signal. So far we have considered basic quantization theory and how modeling can be introduced in this quantization structure. We based our discussion on a mean squared error distortion measure for the speech vector. As discussed in Sect. 14.2.2, the proper measure is the decrease in signal quality as perceived by human listeners. That is, the goal in speech coding is to minimize the perceived degradation resulting from an encoding at a particular rate. This section discusses

methods for integrating perceptually motivated criteria into a coding structure.

To base coding on perceived quality degradation, we must define an appropriate quantitative measure of the perceived distortion. Reasonable objectives for a good distortion measure for a speech codec are a good prediction of experimental data on human perception, mathematical tractability, low delay, and low computational requirements.

A major aspect in the definition of the criterion is the representation of the speech signal the distortion measure operates on. Most straightforward is to quantize the speech signal itself and use the distortion measure as a selection criterion for code vectors and as a means to design the quantizers. This coding structure is commonly used in speech coders based on linear-predictive coding. An alternative coding structure is to apply a transform towards a domain that facilitates a simple distortion criterion. Thus, in this approach, we first perform a mapping to a *perceptual domain* (preprocessing) and then quantize the mapped signal in that domain. At the decoder we apply the inverse mapping (postprocessing). This second architecture is common in transform coders aimed at encoding audio signals at high fidelity.

We start this section with a subsection discussing the squared error criterion, which is commonly used because of its mathematical simplicity. In Subsects. 14.5.2 and 14.5.3 we then discuss models of perception and how the squared error criterion can be used to represent these models. We end the section with a subsection discussing in some more detail the various coding architectures.

14.5.1 Squared Error

The squared-error criterion is commonly used in coding, often without proper physical motivation. Such usage results directly from its mathematical tractability. Given a data sequence, optimization of the model parameters for a model family often leads to a set of linear equations that is easily solved.

For the k-dimensional speech vector $\boldsymbol{x}^k$, the basic squared-error criterion is

$$\eta = (\boldsymbol{x}^k - \hat{\boldsymbol{x}}^k)^{\mathrm{H}}(\boldsymbol{x}^k - \hat{\boldsymbol{x}}^k)\,, \tag{14.45}$$

where the superscript 'H' denotes the Hermitian conjugate and $\hat{\boldsymbol{x}}^k$ is the reconstruction vector upon encoding and decoding. Equation (14.45) quantifies the variance of the signal error. Unfortunately, variance cannot be equated to loudness, which is the psychological correlate of variance. At most we can expect that, for a given original signal, a scaling of the error signal leads to a positive correlation between perceived distortion and squared error.

While the squared error in its basic form is not representative of human perception, adaptive weighting of the squared-error criterion can lead to improved correspondence. By means of weighting we can generalize the squared-error criterion to a form that allows inclusion of knowledge of perception (the formulation of the weighted squared error criterion for a specific perceptual model is described in Subsects. 14.5.2 and 14.5.3). To allow the introduction of perceptual effects, we linearly weight the error vector $\boldsymbol{x}^k - \hat{\boldsymbol{x}}^k$ and obtain

$$\eta = (\boldsymbol{x}^k - \hat{\boldsymbol{x}}^k)^{\mathrm{H}}\mathbf{H}^{\mathrm{H}}\mathbf{H}(\boldsymbol{x}^k - \hat{\boldsymbol{x}}^k)\,, \tag{14.46}$$

where $\mathbf{H}$ is an $m \times k$ matrix, where m depends on the weighting invoked. As we will see below, many different models of perception can be approximated with the simple weighted squared-error criterion of (14.46). In general, the weighting matrix $\mathbf{H}$ adapts to x^k, that is $\mathbf{H}(\boldsymbol{x}^k)$ and

$$\boldsymbol{y}^m = \mathbf{H}(\boldsymbol{x}^k)\,\boldsymbol{x}^k \tag{14.47}$$

can be interpreted as a perceptual-domain representation of the signal vector for a region of $\boldsymbol{x}^k$ where $\mathbf{H}(\boldsymbol{x}^k)$ is approximately constant.

The inclusion of the matrix $\mathbf{H}$ in the formulation of the squared-error criterion generally results in a significantly higher computational complexity for the evaluation of the criterion. Perhaps more importantly, when the weighted criterion of (14.46) is adaptive, then the optimal distribution of the code vectors (Sect. 14.3.3) for constrained-entropy coding is no longer uniform in the speech domain. This has significant implications for the computational effort of a coding system.

The formulation of (14.46) is commonly used in coders that are based on an autoregressive model family, i. e., linear-prediction-based analysis-by-synthesis coding [14.16]. (The matrix $\mathbf{H}$ then usually includes the autoregressive model, as the speech codebook is defined as a filtering of an excitation codebook.) Also in the context of this class of coders, the vector $\mathbf{H}\boldsymbol{x}^k$ can be interpreted as a perceptual-domain vector. However, because $\mathbf{H}$ is a function of $\boldsymbol{x}^k$ it is not straightforward to define a codebook in this domain.

The perceptual weighting matrix $\mathbf{H}$ often represents a filter operation. For a filter with impulse response $[h_0, h_1, h_2, \cdots]$, the matrix $\mathbf{H}$ has a Toeplitz structure:

$$\mathbf{H} = \begin{pmatrix} h_0 & 0 & \cdots \\ h_1 & h_0 & \cdots \\ h_2 & h_1 & \cdots \\ \vdots & \vdots & \ddots \end{pmatrix} . \tag{14.48}$$

For computational reasons, it may be convenient to make the matrix $\mathbf{H}^{\mathrm{H}}\mathbf{H}$ Toeplitz. If the impulse response has time support p then $\mathbf{H}^{\mathrm{H}}\mathbf{H}$ is Toeplitz if $\mathbf{H}$ is selected to have dimension $(m+p)\times m$ [14.19].

Let us consider how the impulse response $[h_0, h_1, h_2, \cdots]$ of (14.48) is typically constructed for the case of linear-predictive coding. The impulse response is constructed from the signal model. Let the transfer function of the corresponding autoregressive model be, as in (14.26)

$$A(z)^{-1} = \frac{\sigma}{1 + a_1 z^{-1} \cdots + a_p z^{-p}} , \tag{14.49}$$

where the a_i are the prediction parameters and σ is the gain. A weighting that is relatively flexible and has low computational complexity is then [14.56]

$$H(z) = \frac{A(z/\gamma_1)}{A(z/\gamma_2)} , \tag{14.50}$$

where γ_1 and γ_2 are parameters that are selected to accurately describe the impact of the distortion on perception. The sequence $[h_0, h_1, h_2, \cdots]$ of (14.48) is now simply the impulse response of $H(z)$. The parameters γ_1 and γ_2 are selected to approximate perception where $1 \geq \gamma_1 > \gamma_2 > 0$. The filter $A(z/\gamma_1)$ deemphasizes the envelope of the power spectral density, which corresponds to decreasing the importance of spectral peaks. The filter $1/A(z/\gamma_2)$ undoes some of this emphasis for a smoothed version of the spectral envelope. The effect is roughly that $1/A(z/\gamma_2)$ limits the spectral reach of the deemphasis $A(z/\gamma_1)$. In other words, the deemphasis of the spectrum is made into a local effect.

To understand coders of the transform model family, it is useful to interpret (14.46) in the frequency domain. We write the discrete Fourier transform (DFT) as the unitary matrix $\mathbf{F}$ and the define a frequency-domain weighting matrix $\mathbf{W}$ such that

$$\mathbf{H}\boldsymbol{x}^k = \mathbf{F}^{\mathrm{H}}\mathbf{W}\mathbf{F}\boldsymbol{x}^k , \tag{14.51}$$

The matrix $\mathbf{W}$ provides a weighting of the frequency-domain vector $\mathbf{F}\boldsymbol{x}^k$. If, for the purpose of our discussion, we neglect the difference between circular and linear convolution and if $\mathbf{H}$ represents a filtering operation (convolution) as in (14.48), then $\mathbf{W}$ is diagonal. To account for perception, we must adapt $\mathbf{W}$ to the input vector $\boldsymbol{x}^k$ (or equivalently, to the frequency-domain vector $\mathbf{F}\boldsymbol{x}^k$) and it becomes a function $\mathbf{W}(\boldsymbol{x}^k)$: Equation (14.50) could be used as a particular mechanism for such weighting. However, in the transform coding context, so-called *masking* methods, which are described in Sect. 14.5.2, are typically used to find $W(\boldsymbol{x}^k)$.

As mentioned before, the random vector $\boldsymbol{Y}^m = \mathbf{H}\boldsymbol{X}^k$ (or, equivalently, the vector $\mathbf{W}\mathbf{F}\boldsymbol{X}^k$) can be considered as a perceptual-domain description. Assuming smooth behavior of $\mathbf{H}(\boldsymbol{x}^k)$ as a function of $\boldsymbol{x}^k$, this domain can then be used as the domain for coding. A codebook must be defined for the perceptual-domain vector $\boldsymbol{Y}^m$ and we select entries from this codebook with the unweighted squared error criterion. This approach is common in transform coding. When this coding in the perceptual domain is used, the distortion measure does not vary with the vector $\boldsymbol{y}^m$, and a uniform quantizer is optimal for $\boldsymbol{Y}^m$ for the constrained-entropy case. If the mapping to the perceptual domain is unique and invertible (which is not guaranteed by the formulation), then $\boldsymbol{y}^m = \mathbf{H}(\boldsymbol{x}^k)\boldsymbol{x}^k$ ensures that $\boldsymbol{x}^k$ is specified when $\boldsymbol{y}^m$ is known and only indices to the codebook for $\boldsymbol{Y}^m$ need to be encoded. In practice, the inverse mapping may not be unique, resulting in problems at block boundaries and the inverse may be difficult to compute. As a result it is common practice to quantize and transmit the weighting $\mathbf{W}$, e.g., [14.57, 58].

14.5.2 Masking Models and Squared Error

Extensive quantitative knowledge of auditory perception exists and much of the literature on quantitative descriptions of auditory perception relates to the concept of *masking*, e.g., [14.59–63]. The masking-based description of the operation of the auditory periphery can be used to include the effect of auditory perception in speech and audio coding. Let us define an arbitrary signal that we call the *masker*. The masker implies a set of second signals, called *maskees*, which are defined as signals that are not audible when presented in the presence of the masker. That is, the maskee is below the *masking threshold*. Masking explains, for example, why a radio must be made louder in a noisy environment such as a car. We can think of masking as being

a manifestation of the internal precision of the auditory periphery.

In general, laws for the masking threshold are based on psychoacoustic measurements for the masker and maskee signals that are constructed independently and then added. However, it is clear that the coding error is correlated to the original signal. In the context of masking it is a commonly overlooked fact that, for ideal coding, the coding error signal is, under certain common conditions, independent of the reconstructed signal [14.12]. Thus, a reasonable objective of audio and speech coding is to ensure that the coding error signal is below the masking threshold of the *reconstructed* signal.

Masking is quantified in terms of a so-called masking curve. We provide a generalized definition of such a curve. Let us consider a signal vector $\boldsymbol{x}^k$ with k samples that is defined in $\mathbb{R}^k$. We define a perceived-error measurement domain by any invertible mapping $\mathbb{R}^k \to \mathbb{R}^m$. Let $\{\boldsymbol{e}_i^m\}_{i\in\{0,\cdots,m-1\}}$ be the unit-length basis vectors that span $\mathbb{R}^m$. We then define the m-dimensional *masking curve* as [14.64] JND_i, $i \in 0, \cdots, m-1$, where the scalar JND_i is the *just-noticeable difference* (JND) for the basis vector $\boldsymbol{e}_i^m$. That is, the vector $JND_i\,\boldsymbol{e}_i^m$ is precisely at the threshold of being audible for the given signal vector.

Examples of the masking curve can be observed in the time and the frequency domain. The frequency-domain representation of $\boldsymbol{x}^k$ is $\mathbf{F}\boldsymbol{x}^k$. *Simultaneous* masking is defined as the masking curve for $\mathbf{F}\boldsymbol{x}^k$, i. e., the just-noticeable amplitudes for the frequency unit vectors $\boldsymbol{e}_1$, $\boldsymbol{e}_2$, etc. In the time domain we refer to *nonsimultaneous* (or *forward* and *backward*) masking depending on whether the time index i of the unit vectors $\boldsymbol{e}_i$ is prior to or after the main event in the masker (e.g., an onset). Both the time-domain (temporal) and frequency-domain masking curves are asymmetric and dependent on the loudness of the masker. A loud sound leads to a rapid decrease in auditory acuity, followed by a slow recovery to the default level. The recovery may take several hundreds of ms and causes forward masking. The decrease in auditory acuity before a loud sound, backward masking, extends only over very short durations (at most a few ms). Similar asymmetry occurs in the frequency domain, i. e., in simultaneous masking. Let us consider a tone. The auditory acuity is decreased mostly at frequencies higher than the tone. The acuity increases more rapidly from the masker when moving towards lower frequencies than when moving towards higher frequencies, which is related to the decrease in frequency resolution with increasing frequency. A significant difference exists in the masking between tonal and noise-like signals. We refer to [14.63, 65, 66] for further information on masking.

The usage of masking is particularly useful for coding in the perceptual domain with a constraint that the quality is to be transparent (at least according to the perceptual knowledge provided). For example, consider a transform coder (based on either the discrete cosine transform or the DFT). In this case, the quantization step size can be set to be the JND as provided by the simultaneous masking curve [14.57].

Coders are commonly subject to a bit-rate constraint, which means knowledge of the masking curve is not sufficient. A distortion criterion must be defined based on the perceptual knowledge given. A common strategy in audio coding to account for simultaneous masking is to use a weighted squared error criterion, with a diagonal weighting matrix $\mathbf{H}$ that is reciprocal of the masking threshold [14.27, 58, 67–70]. In fact, this is a general approach that is useful to convert a masking curve in any measurement domain:

$$\mathbf{H} = \begin{pmatrix} \frac{1}{JND_1} & 0 & \cdots \\ 0 & \frac{1}{JND_2} & \cdots \\ \vdots & \vdots & \ddots \end{pmatrix} , \tag{14.52}$$

where it is understood that the weighting matrix $\mathbf{H}$ is defined in the measurement domain. To see this consider the effect of the error vector $JND_i\,\boldsymbol{e}_i^m$ on the squared error:

$$\begin{aligned} \eta &= JND_i^2\,\boldsymbol{e}_i^{m\mathrm{H}}\mathbf{H}^{\mathrm{H}}\mathbf{H}\boldsymbol{e}_i^m \\ &= JND_i^2\;JND_i^{-2} = 1 \,. \end{aligned} \tag{14.53}$$

Thus, the points on the masking curve are defined as the amplitudes of basis vectors that lead to a unit distortion. This is a reasonable motivation for the commonly used reciprocal-weighting approach for the squared-error criterion defined by the weighting described in (14.52). However, it should be noted that for this formulation the distortion measure does not vanish below the masking threshold. A more-complex approach where the distortion measure does vanish below the JND is given in [14.71].

14.5.3 Auditory Models and Squared Error

The weighting procedure of (14.52) (possibly in combination with the transform to the measurement domain)

is an operation that transforms the signal to a perceptually relevant domain. Thus, the operation can be interpreted as a simple auditory model. Sophisticated models of the auditory periphery that directly predict the input to the auditory nerve have also been developed, e.g., [14.72–77]. Despite the existence of such quantitative models of perception, their application in speech coding has been limited. Only a few examples [14.78,79] of the explicit usage of existing quantitative knowledge of auditory perception in speech coding speech exist. In contrast, in the field of audio coding the usage of quantitative auditory knowledge is common. Transform coders can be interpreted as methods that perform coding in the perceptual domain, using a simple perceptual model, usually based on (simultaneous) masking results.

We can identify a number of likely causes for the lack of usage of auditory knowledge in speech coding. First, the structure of speech coders and the constraint on computational complexity naturally leads to speech-coding-specific models of auditory perception, such as (14.50). The parameters of these simple speech-coding-based models are optimized directly based on coding performance. Second, the perception of the periodicity nature of voiced speech, often referred to as the perception of *pitch*, is not well understood in a quantitative manner. It is precisely the distortion associated with the near-periodic nature of voiced speech that is often critical for the perceived quality of the reconstructed signal. An argument against using a quantitative model based on just-noticeable differences (JNDs) is that JNDs are often exceeded significantly in speech coding. While the weighting of (14.52) is reasonable near the JND threshold value, it may not be accurate in the actual operating region of the speech coder. Major drawbacks of using sophisticated models based on knowledge of the auditory periphery are that they tend to be computationally expensive, have significant latency, and often lead to a representation that has many more dimensions than the input signal. Moreover the complexity of the model structure makes inversion difficult, although not impossible [14.80].

The complex structure of auditory models that describe the functionality of the auditory periphery is time invariant. We can replace it by a much simpler structure at the cost of making it time variant. That is, the mapping from the speech domain to the perceptual domain can be simplified by approximating this mapping as locally linear [14.79]. Such an approximation leads to the *sensitivity matrix approach*, which was first introduced in a different context by [14.53] and described in a rigorous general manner in [14.81]. If a mapping from the speech domain vector $\boldsymbol{x}^k$ to an auditory domain vector $\boldsymbol{y}^m$ (as associated with a particular model of the auditory periphery) can be approximated as locally linear, then for a small coding error $\boldsymbol{x}^k - \hat{\boldsymbol{x}}^k$, we can write $\boldsymbol{y}^m - \hat{\boldsymbol{y}}^m$ as a matrix multiplication of $\boldsymbol{x}^k - \hat{\boldsymbol{x}}^k$:

$$\boldsymbol{y}^m - \hat{\boldsymbol{y}}^m \approx \mathbf{H}\left(\boldsymbol{x}^k - \hat{\boldsymbol{x}}^k\right), \tag{14.54}$$

where $\mathbf{H}$ is an $m \times k$ matrix. This means that, for the set of codebook vectors from $C_{\boldsymbol{X}^k}$ that is sufficiently close to $\boldsymbol{x}^k$, the squared-error criterion of (14.46) forms an approximation to the psychoacoustic measure. Moreover, selecting the nearest codebook entry from $C_{\boldsymbol{X}^k}$ using (14.46) results in the globally optimal codebook vector for the input vector $\boldsymbol{x}^k$. In the sensitivity matrix approach, the first step for each speech vector $\boldsymbol{x}^k$ is to find the $k \times k$ *sensitivity matrix* $\mathbf{H}^{\mathrm{T}}\mathbf{H}$. This operation is based on an analysis of the distortion criterion [14.79]. Once this has been done, the selection of the codebook entries is similar to that for a signal-invariant weighted squared-error distortion measure.

In the sensitivity matrix approach, the matrix $\mathbf{H}$ is a function of the past and future signal:

$$\mathbf{H} = \mathbf{H}(\cdots, \boldsymbol{x}_{i-1}^k, \boldsymbol{x}_i^k, \boldsymbol{x}_{i+1}^k, \cdots), \tag{14.55}$$

where $\boldsymbol{x}_i^k$ is the current speech vector, $\boldsymbol{x}_{i-1}^k$ is the previous speech vector, etc. To avoid the introduction of latency, the future speech vectors can be replaced by a prediction of these vectors from the present and past speech vectors.

The sensitivity matrix approach requires that the mapping from speech domain to perceptual domain is continuous and differentiable, which is not the case for psychoacoustic models. The approximation of such discontinuities by continuous functions generally leads to satisfactory results.

The sensitivity matrix approach is well motivated in the context of a speech-domain codebook $C_{\boldsymbol{X}^k}$ and a criterion that consists of a perceptual transform followed by the squared error criterion. The search through the speech-domain codebook with a perceptual criterion then reduces to searching with a weighted squared-error criterion.

The benefit of the sensitivity matrix approach is not so obvious if the signal vector codebook, $C_{\boldsymbol{X}^k}$, is defined in the perceptual domain. However, it can be useful if the perceptual transform is known to the decoder (by, for example, transmission of an index). The perceptual domain often has higher dimensionality than the corresponding

speech block. The singular-value decomposition of $\mathbf{H}$ can then be used to reduce the dimensionality of the perceptual domain error vector to k. Let $\mathbf{H} = \mathbf{VDU}$ be a singular value decomposition, where $\mathbf{V}$ is an $m \times m$ unitary matrix, $\mathbf{D}$ is a $m \times k$ diagonal matrix and $\mathbf{U}$ is a unitary $k \times k$ matrix

14.5.4 Distortion Measure and Coding Architecture

As we have seen, the distortion measure has a significant impact on the architecture of a speech coder. In this subsection, we summarize the above discussion from a codec-architecture viewpoint.

Speech-Domain Codebook

The most straightforward architecture for a speech coder is to define the codebook in the speech domain and use an appropriate distortion measure during encoding and during training of the codebook. In general, the measure is adaptive. This approach, which is shown in Fig. 14.1, is most common in speech coding. An advantage is that the decoder does not require knowledge of the time-varying distortion measure.

We saw in Sect. 14.3.4 that it can be advantageous to use speech codebooks that are associated with models. This simplifies the codebook structure and, in the case of constrained-resolution coding, its size. The codebooks can be generated in real time as, for example, in the case of linear-prediction (autoregressive model)-based analysis-by-synthesis coding [14.16, 55].

The underlying aim of the speech-domain codebook architecture is generally to approximate auditory perception by an adaptively weighted squared-error criterion. Usually, this criterion is heuristic and based on tuning within the context of the coder (as is typically done for (14.50)). However, as shown in Sect. 14.5.3, the sensitivity analysis method facilitates the usage of complex auditory models. This approach requires, at least in principle, no further tuning.

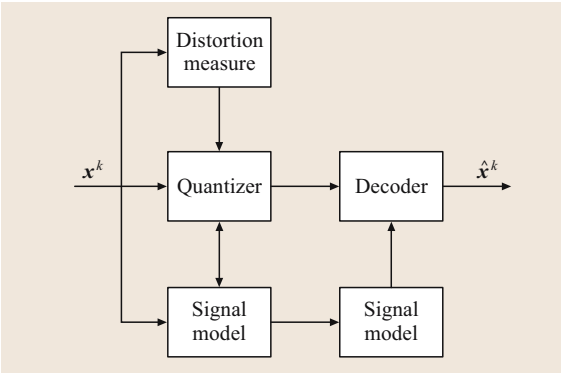

Fig. 14.1 Common architecture for coding with a distortion measure. A signal quantization index and a signal model index are transmitted

Disadvantages of the weighted squared-error criterion are its computational complexity and that, in contrast to the unweighted squared-error measure, it does not lead to uniform codebooks for the constrained-entropy case. In the context of autoregressive-model-based analysis-by-synthesis coding, many procedures have been developed to reduce the computational complexity of the weighted squared-error criterion [14.18, 19, 82].

Perceptual-Domain Codebook

As an alternative to defining the codebook in the speech domain, we can define the codebook in a perceptual domain, as is shown in Fig. 14.2. We define a perceptual domain as a domain where the unweighted squared error criterion can be applied. The most elegant paradigm requires no information about the speech vector other than its index in the perceptual domain codebook. This elegance applies if the mapping to the perceptual domain is injective (one to one). Then if $\boldsymbol{y}^m$ is known, $\boldsymbol{x}^k$ is also known. An example is an auditory model that is a weighted DFT or DCT with a one-to-one function $\mathbb{R}^k \to \mathbb{R}^k$ that maps $\boldsymbol{x}^k$ into $\boldsymbol{y}^k$. An inverse func-

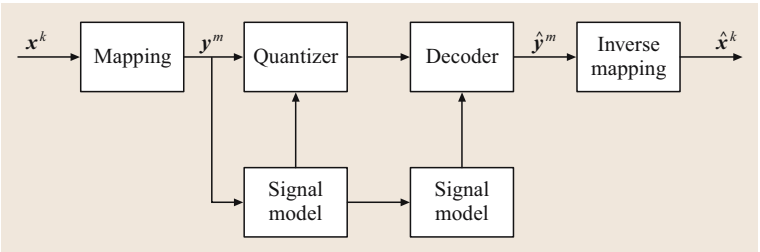

Fig. 14.2 Ideal architecture for coding in the perceptual domain, with invertible mapping. A perceptual-domain quantization index and a signal model index are transmitted. The signal model can be omitted

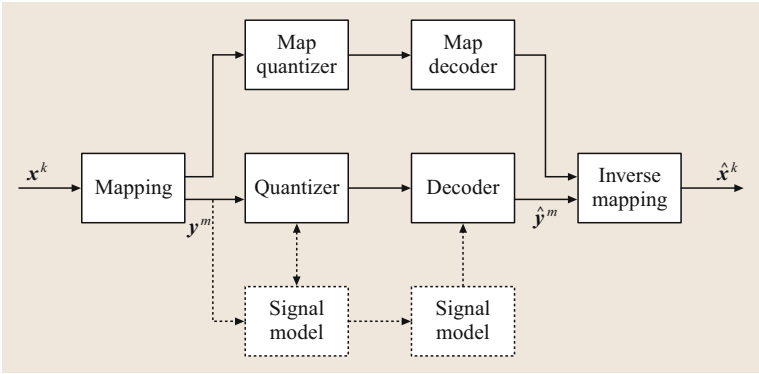

Fig. 14.3 Architecture for coding in the perceptual domain with encoded mapping. This architecture is common in transform coding. A mapping index, a perceptual-domain quantization index, and a signal model index are transmitted. The signal model is commonly omitted

tion can be derived by the decoder from the quantized vector $\hat{\boldsymbol{y}}^{\mathrm{k}}$.

The non-uniqueness of the auditory mapping from the speech domain to the auditory domain results in practical problems. Particularly if the models are not accurate, the nonuniqueness can result in mismatches between coding blocks and severe audible distortion.

The uniqueness issue for the mapping can be solved, at the cost of increased rate, by transmitting information about the mapping as is shown in Fig. 14.3. For example, for transform coders used for audio signals, the masking curve is commonly transmitted, e.g., [14.52, 58, 67–69]. To reduce the rate required, this is commonly done on a per-frequency-band basis. The bands generally are uniformly spaced on an equivalent rectangular bandwidth (ERB) or mel scale. In practice the masking curve is not always transmitted directly, but a maximum amplitude, and the number of quantization levels for each band are transmitted, e.g., [14.27, 68, 70].

14.6 Summary

This chapter discussed the principles underlying the transmission of speech (and audio) signals. The main attributes of coding, rate, quality, robustness to channel errors, delay, and computational complexity were discussed first. We then provided a generic perspective of speech coding.

Each block of speech samples was described as a random vector. Information about the vector was transmitted in the form of a codebook index. The relation between the reconstruction vector density and the data density was given. We then modeled the probability density of the speech vector as a weighted sum of component probability density functions, each describing the speech vector probability density for a particular speech sound. Each such component density function corresponds to a *model*. In the case of linear-prediction-based (equivalent to autoregressive-model-based) coding, each component function (and thus model) is characterized by a set of predictor parameters. The approach results in the standard two-step speech coding approach, in which we first extract the model and then code the speech vector given the model. We showed that this approach leads to standard distortion measures used for the quantization of the predictor parameters.

We showed that the number of models (component probability density functions) is independent of the overall coding rate. Thus, the rate spent on linear-prediction parameters in linear-predictive coding should not vary with rate (at least when the rate is high). It was shown that practical coders indeed have this behavior. We emphasized also that the rate allocated to the model (the predictor parameters) is not a direct function of a perceptual threshold, but the result of an optimal trade-off between the rate allocated for the speech given the model and the rate allocated for the model. We showed that it is possible to calculate the rate allocation for the model (the bit allocation for the predictor parameters) and that the result provided is close to practical codec configurations. We discussed analysis-by-synthesis coding as a particular application of the two-stage coding method. We noted that analysis-by-synthesis coding is not optimal because the speech-domain codebook has suboptimal quantization cell shapes.

Finally we discussed how perception can be integrated into the coding structure. We distinguished coding in a perceptually relevant domain, which is commonly used in audio coding, from coding in the speech-signal domain, which is commonly used in speech coding. The advantage of coding in the perceptual domain is that a simple squared-error criterion can be used. However, in practice the method generally requires some form of encoding of the mapping, so that the decoder can perform an inverse mapping. Improved mapping procedures may change this requirement.

References

14.1 W.B. Kleijn, K.K. Paliwal: An introduction to speech coding. In: *Speech Coding and Synthesis*, ed. by W.B. Kleijn, K.K. Paliwal (Elsevier, Amsterdam 1995) pp. 1–47

14.2 R.V. Cox: Speech coding standards. In: *Speech Coding and Synthesis*, ed. by W.B. Kleijn, K.K. Paliwal (Elsevier, Amsterdam 1995) pp. 49–78

14.3 R. Salami, C. Laflamme, J. Adoul, A. Kataoka, S. HAyashi, T. Moriya, C. Lamblin, D. Massaloux, S. Proust, P. Kroon, Y. Shoham: Design and description of CS-ACELP: a toll quality 8 kb/s speech coder, IEEE Trans. Speech Audio Process. **6**(2), 116–130 (1998)

14.4 B. Bessette, R. Salami, R. Lefebvre, M. Jelinek, J. Rotola-Pukkila, J. Vainio, H. Mikkola: The adaptive multirate wideband speech codec (amr-wb), IEEE Trans. Speech Audio Process. **6**(8), 620–636 (2002)

14.5 ITU-T Rec. P.800: *Methods for Subjective Determination of Transmission Quality* (1996)

14.6 A.W. Rix: Perceptual speech quality assessment – a review, Proc. IEEE ICASSP, Vol. 3 (2004) pp. 1056–1059

14.7 S. Möller: *Assessment and Prediction of Speech Quality in Telecommunications* (Kluwer Academic, Boston 2000)

14.8 P. Kroon: Evaluation of speech coders. In: *Speech Coding and Synthesis*, ed. by W.B. Kleijn, K.K. Paliwal (Elsevier, Amsterdam 1995) pp. 467–493

14.9 W. Stallings: *High-speed networks: TCP/IP and ATM design principles* (Prentice Hall, Englewood Cliffs 1998)

14.10 Information Sciences Institute: Transmission control protocol, IETF RFC793 (1981)

14.11 J. Postel: User datagram protocol, IETF RFC768 (1980)

14.12 T.M. Cover, J.A. Thomas: *Elements of Information Theory* (Wiley, New York 1991)

14.13 N. Kitawaki, K. Itoh: Pure delay effects on speech quality in telecommunications, IEEE J. Sel. Area. Comm. **9**(4), 586–593 (1991)

14.14 J. Cox: The minimum detectable delay of speech and music, Proc. IEEE ICASSP, Vol. 1 (1984) pp. 136–139

14.15 J. Chen: A robust low-delay CELP speech coder at 16 kb/s. In: *Advances in Speech Coding*, ed. by B.S. Atal, V. Cuperman, A. Gersho (Kluwer Academic, Dordrecht 1991) pp. 25–35

14.16 B.S. Atal, M.R. Schroeder: Stochastic coding of speech at very low bit rates, Proc. Int. Conf. Comm. (1984) pp. 1610–1613

14.17 J.-P. Adoul, P. Mabilleau, M. Delprat, S. Morisette: Fast CELP coding based on algebraic codes, Proc. IEEE ICASSP (1987) pp. 1957–1960

14.18 I.M. Trancoso, B.S. Atal: Efficient procedures for selecting the optimum innovation in stochastic coders, IEEE Trans. Acoust. Speech **38**(3), 385–396 (1990)

14.19 W.B. Kleijn, D.J. Krasinski, R.H. Ketchum: Fast methods for the CELP speech coding algorithm, IEEE Trans. Acoust. Speech **38**(8), 1330–1342 (1990)

14.20 T. Lookabough, R. Gray: High-resolution theory and the vector quantizer advantage, IEEE Trans. Inform. Theory **IT-35**(5), 1020–1033 (1989)

14.21 S. Na, D. Neuhoff: Bennett's integral for vector quantizers, IEEE Trans. Inform. Theory **41**(4), 886–900 (1995)

14.22 S.P. Lloyd: Least squares quantization in PCM, IEEE Trans. Inform. Theory **IT-28**, 129–137 (1982)

14.23 Y. Linde, A. Buzo, R.M. Gray: An algorithm for vector quantizer design, IEEE Trans. Commun. **COM-28**, 84–95 (1980)

14.24 P. Chou, T. Lookabough, R. Gray: Entropy-constrained vector quantization, IEEE Trans. Acoust. Speech **38**(1), 31–42 (1989)

14.25 A. Gersho: Asymptotically optimal block quantization, IEEE Trans. Inform. Theory **25**, 373–380 (1979)

14.26 P. Swaszek, T. Ku: Asymptotic performance of unrestricted polar quantizers, IEEE Trans. Inform. Theory **32**(2), 330–333 (1986)

14.27 R. Vafin, W.B. Kleijn: Entropy-constrained polar quantization and its application to audio coding, IEEE Trans. Speech Audio Process. **13**(2), 220–232 (2005)

14.28 J.J. Rissanen, G. Langdon: Arithmetic coding, IBM J. Res. Devel. **23**(2), 149–162 (1979)

14.29 J. Rissanen: Modeling by the shortest data description, Automatica **14**, 465–471 (1978)

14.30 J. Rissanen: A universal prior for integers and estimation by minimum description length, Ann. Stat. **11**(2), 416–431 (1983)

14.31 P. Grunwald: A tutorial introduction to the minimum description length principle. In: *Advances in Minimum Description Length: Theory and Applications*, ed. by P. Grunwald, I.J. Myung, M. Pitt (MIT, Boston 2005)

14.32 A. Barron, T.M. Cover: Minimum complexity density estimation, IEEE Trans. Inform. Theory **37**(4), 1034–1054 (1991)

14.33 A.H. Gray, J.D. Markel: Distance measures for speech process, IEEE Trans. Acoust. Speech Signal Process. **ASSP-24**(5), 380–391 (1976)

14.34 R. Hagen, P. Hedelin: Low bit-rate spectral coding in CELP a new LSP method, Proc. IEEE ICASSP (1990) pp. 189–192

14.35 K.K. Paliwal, B.S. Atal: Efficient vector quantization of LPC parameters at 24 bits/frame, IEEE Trans. Speech Audio Process. **1**(1), 3–14 (1993)

14.36 C. Xydeas, C. Papanastasiou: Split matrix quantization of lpc parameters, IEEE Trans. Speech Audio Process. **7**(2), 113–125 (1999)

14.37 A. Subramaniam, B. Rao: Speech LSF quantization with rate independent complexity, bit scalability, and learning, Proc. IEEE ICASSP (2001) pp. 705–708

14.38 U. Grenander, G. Szego: *Toeplitz Forms and their Applications* (Chelsea, New York 1984)

14.39 F. Itakura, S. Saito: Speech information compression based on the maximum likelihood estimation, J. Acoust. Soc. Jpn. **27**(9), 463 (1971)

14.40 S. Saito, K. Nakata: *Fundamentals of Speech Signal Process* (Academic, New York 1985)

14.41 P.J. Brockwell, R.A. Davis: *Time Series: Theory and Methods* (Springer, New York 1996)

14.42 F. Itakura, S. Saito: Analysis Synthesis Telephony Based Upon the Maximum Likelihood Method, Reports of 6th Int. Cong. Acoust.,C-5-5, C17-20, ed. by Y. Kohasi (1968)

14.43 R.M. Gray, A. Buzo, A.H. Gray, Y. Matsuyama: Distortion measures for speech process, IEEE Trans. Acoust. Speech Signal Process. **ASSP-28**(4), 367–376 (1980)

14.44 K.K. Paliwal, W.B. Kleijn: Quantization of LPC parameters. In: *Speech Coding and Synthesis*, ed. by W.B. Kleijn, K.K. Paliwal (Elsevier, Amsterdam 1995) pp. 433–466

14.45 W.R. Gardner, B.D. Rao: Noncausal all-pole modeling of voiced speech, IEEE Trans. Speech Audio Process. **5**(1), 1–10 (1997)

14.46 M. Nilsson, W.B. Kleijn: Shannon entropy estimation based on high-rate quantization theory, Proc. EUSIPCO (2004) pp. 1753–1756

14.47 M. Nilsson: *Entropy and Speech* (Royal Institute of Technology, Stockholm 2006), Ph.D. dissertation, KTH

14.48 C. Lamm: *Improved Spectral Estimation in Speech Coding* (Lund Institute of Technology (LTH), Lund 1998), Master's thesis

14.49 K.L.C. Chan: Split-dimension vector quantization of parcor coefficients for low bit rate speech coding, IEEE Trans. Speech Audio Process. **2**(3), 443–446 (1994)

14.50 A. Subramaniam, B.D. Rao: PDF optimized parametric vector quantization of speech line spectral freuencies, IEEE Speech Coding Workshop (Delavan 2000) pp. 87–89

14.51 P. Hedelin, J. Skoglund: Vector quantization based on Gaussian mixture models, IEEE Trans. Speech Audio Process. **8**(4), 385–401 (2000)

14.52 S. Srinivasan, J. Samuelsson, W.B. Kleijn: Speech enhancement using a-priori information with classified noise codebooks, Proc. EUSIPCO (2004) pp. 1461–1464

14.53 W.R. Gardner, B.D. Rao: Optimal distortion measures for the high rate vector quantization of LPC parameters, Proc. IEEE ICASSP (1995) pp. 752–755

14.54 M.Y. Kim, W.B. Kleijn: KLT-based adaptive classified vector quantization of the speech signal, IEEE Trans. Speech Audio Process. **12**(3), 277–289 (2004)

14.55 P. Kroon, E.F. Deprettere: A class of analysis-by-synthesis predictive coders for high quality speech coding at rates between 4.8 and 16 kbit/s, IEEE J. Sel. Area. Commun. **6**(2), 353–363 (1988)

14.56 J. Chen, A. Gersho: Real-time vector APC speech coding at 4–800 bps with adaptive postfiltering, Proc. IEEE ICASSP (1987) pp. 2185–2188

14.57 J. Johnston: Transform coding of audio signals using perceptual noise criteria, IEEE J. Sel. Area. Commun. **6**(2), 314–323 (1988)

14.58 H. Malvar: Enhancing the performance of subband audio coders for speech signals, Proc. IEEE Int. Symp. on Circ. Syst., Vol. 5 (1998) pp. 98–101

14.59 R. Veldhuis: Bit rates in audio source coding, IEEE J. Sel. Area. Commun. **10**(1), 86–96 (1992)

14.60 B.C.J. Moore: Masking in the human auditory system. In: *Collected papers on digital audio bit-rate reduction*, ed. by N. Gilchrist, C. Grewin (Audio Eng. Soc., New York 1996)

14.61 B.C.J. Moore: *An Introduction to the Psychology of Hearing* (Academic, London 1997)

14.62 E. Zwicker, H. Fastl: *Psychoacoustics* (Springer Verlag, Berlin, Heidelberg 1999)

14.63 T. Painter, A. Spanias: Perceptual coding of digital audio, Proc. IEEE **88**(4), 451–515 (2000)

14.64 J.H. Plasberg, W.B. Kleijn: The sensitivity matrix: Using advanced auditory models in speech and audio processing, IEEE Trans. Speech Audio Process. **15**, 310–319 (2007)

14.65 J.L. Hall: Auditory psychophysics for coding applications. In: *The Digital Signal Processing Handbook*, ed. by V.K. Madisetti, D. Williams (CRC, Boca Raton 1998) pp. 39.1–39.25

14.66 W. Jesteadt, S.P. Bacon, J.R. Lehman: Forward masking as a function of frequency, masker level and signal delay, J. Acoust. Soc. Am. **71**(4), 950–962 (1982)

14.67 D. Sinha, J.D. Johnston: Audio compression at low bit rates using a signal adaptive switched

filterbank, Proc. IEEE ICASSP, Vol. 2 (1996) pp. 1053–1056
14.68 T. Verma, T. Meng: A 6 kbps to 85 kbps scalable audio coder, Proc. IEEE ICASSP, Vol. 2 (2000) pp. II877–II880
14.69 A.S. Scheuble, Z. Xiong: Scalable audio coding using the nonuniform modulated complex lapped transform, Proc. IEEE ICASSP, Vol. 5 (2001) pp. 3257–3260
14.70 R. Heusdens, R. Vafin, W.B. Kleijn: Sinusoidal modeling using psychoacoustic-adaptive matching pursuits, IEEE Signal Proc. Lett. **9**(8), 262–265 (2002)
14.71 M.Y. Kim, W.B. Kleijn: Resolution-constrained quantization with JND based perceptual-distortion measures, IEEE Signal Proc. Lett. **13**(5), 304–307 (2006)
14.72 O. Ghitza: Auditory nerve representation as a basis for speech processing. In: *Advances in Speech Signal Processing* (Dekker, New York 1992) pp. 453–485
14.73 T. Dau, D. Püschel, A. Kohlrausch: A quantitative model of the effective signal processing in the auditory system. I. Model structure, J. Acoust. Soc. Am. **99**(6), 3615–3622 (1996)
14.74 T. Dau, B. Kollmeier, A. Kohlrausch: Modeling auditory processing of amplitude modulation. I. detection and masking with narrowband carriers, J. Acoust. Soc. Am. **102**(5), 2892–2905 (1997)
14.75 G. Kubin, W.B. Kleijn: On speech coding in a perceptual domain, Proc. IEEE ICASSP, Vol. I (1999) pp. 205–208
14.76 F. Baumgarte: *Ein psychophysiologisches Gehörmodell zur Nachbildung von Wahrnehmungsschwellen für die Audiocodierung* (Univ. Hannover, Hannover 2000), Ph.D. dissertation (in German)
14.77 S. van de Par, A. Kohlrausch, G. Charestan, R. Heusdens: A new psychoacoustical masking model for audio coding applications, Proc. IEEE ICASSP (2002) pp. 1805–1808
14.78 D. Sen, D. Irving, W. Holmes: Use of an auditory model to improve speech coders, Proc. IEEE ICASSP (1993) pp. II411–II414
14.79 J.H. Plasberg, D.Y. Zhao, W.B. Kleijn: The sensitivity matrix for a spectro-temporal auditory model, Proc. EUSIPCO (2004) pp. 1673–1676
14.80 X. Yang, K. Wang, S. Shamma: Auditory representation of acoustic signals, IEEE Trans. Inform. Theory **38**(2), 824–839 (1996)
14.81 T. Linder, R. Zamir, K. Zeger: High-resolution source coding for non-difference measures: the rate-distortion function, IEEE Trans. Inform. Theory **45**(2), 533–547 (1999)
14.82 I. Gerson, M. Jasiuk: Vector sum excited linear prediction (VSELP), Proc. IEEE ICASSP (1990) pp. 461–464