5

Review of MPEG–4 General Audio Coding

James D. Johnston and Schuyler R. Quackenbush
AT&T Labs, Florham Park, New Jersey

Jürgen Herre and Bernhard Grill
Fraunhofer Gesellschaft IIS, Erlangen, Germany

I. INTRODUCTION

Inside the MPEG–4 standard, there are various ways to encode or describe audio. They can be grouped into two basic kinds of audio services, called general audio coding and synthetic audio. General audio coding is concerned with taking pulse code–modulated (PCM) audio streams and efficiently encoding them for transmission and storage; synthetic audio involves the synthesis, creation, and parametric description of audio signals. In this chapter, we will discuss general audio coders.

A general audio coder operates in the environment of Figure 1. The encoder’s input is a PCM stream. The encoder creates a bitstream that can either be decoded by itself or inserted into an MPEG–4 systems layer. The resulting bitstream can be either stored or transmitted. At the decoder end, the bitstream, after being demultiplexed from any system layer, is converted back into a PCM stream representing the audio stream at the input.

A. Kinds of Coders

Historically, most audio coders (including speech coders) have attempted to extract redundancy in order to avoid transmitting bits that do not convey information. Some examples of these coders, in quasi-historical order, are LPC (linear predictive coding) [1], DPCM (differential pulse-code modulation) [1], ADPCM (adaptive differential pulse-code modulation) [1], subband coding [2], transform coding [1], and CELP (codebook excited linear prediction) [3]. All of these coders use a model of the source in one fashion or another in order to reduce the bit rate and are called, accordingly, source coders. These source coders can, in general, be either lossless (i.e., they return exactly the same signal that was input to the coder) or lossy, but they are in general lossy.

1. Source Coding

Source coding is a very good way to reduce bit rate if (1) the material being coded has a good source model and (2) the source redundancy allows sufficient coding gain to pro-
provide the required compression ratio. However, in the case of music, source models can be essentially arbitrary and can change abruptly (unlike speech, for which the model changes are limited by the physics of the vocal tract). In addition, the compression ratios required for efficient transmission may require more gain than a pure source coder can manage.

2. **Perceptual Coding**

As channel rates become lower and lower and source models no longer provide sufficient gain or become unstable, something more than source coding becomes necessary. In an audio signal, many parts of the signal are not actually audible [4–8]. They are "masked" by other parts of the signal or are below the absolute threshold of hearing. There is no need to send the parts of the signal that are inaudible if the signal will not be processed substantially after decoding. This is the principal means of compression in the perceptual coder. Unlike a source coder, the perceptual coder is a kind of destination coder, where the destination (i.e., human auditory system) is considered, and parts of the signal that are irrelevant are discarded. This active removal of irrelevance is the defining characteristic of the perceptual coder.

B. **MPEG General Audio Coding**

There are various audio coding methods in MPEG, from pure source coding techniques such as CELP to sophisticated perceptual algorithms such as advanced audio coding (AAC). The encoding techniques follow three basic block diagrams, shown in Figures 2, 3, 4, corresponding to a general LPC, a subband coder, and a perceptual coder, respectively.

In Figure 2, we show a generalized block diagram of an LPC encoder. There are many kinds of LPC encoders, hence the block diagram has a number of required and optional parts. Bear in mind that a CELP coder is a kind of LPC encoder, using a vector quantization strategy. In the diagram, the three blocks outlined by solid lines are the required blocks, i.e., a difference calculator, a quantizer, and a predictor. These three blocks constitute the basic LPC encoder. The heart of the LPC coder is the predictor, which uses the history of the encoded signal in order to build an estimate of what the next sample is. This predicted signal is subtracted from the next sample; the result, called the error signal, is then quantized; and the quantized value is stored or transmitted. The decoder is quite simple; it simply decodes the error signal, adds it to the predictor output, and puts that signal back into the predictor. If noise shaping is present, inverse noise shaping would also be applied before the signal is output from the decoder. The gain in this coder comes
Figure 2 A general block diagram of an LPC coder.

Figure 3 A general block diagram of a subband coder.

Figure 4 A general block diagram of a perceptual coder.
from the reduction in energy of the error signal, which results in a decrease in total quantization noise. Although the pitch predictor is not shown as optional, some LPC coders omit it completely or incorporate it in the LPC predictor.

The optional blocks are shown with dashed lines. These are the noise shaping filter, which can be fixed (like a fixed predictor) or adapted in a signal-dependent fashion, and the two adaptation blocks, corresponding to forward and backward adaptation. The control signals resulting from forward adaptation are shown using a dot-dashed line and those from the backward adaptation with a dotted line.

Most modern LPC coders are adaptive. As shown in the block diagram, there are two kinds of adaptation, backward and forward. The difference is that forward adaptation works on the input signal and must transmit information forward to the decoder for the decoder to recover the signal, whereas backward adaptation works on information available to the decoder and no information need be expressly transmitted. Unfortunately, the comparison is not that simple, as the response time of a backward-adaptive system must lag the signal because it can only look at the transmitted signal, whereas the forward-adaptive system can use delay to look ahead to track the signal, adapt quickly, and adapt without having to compensate for the presence of coding noise in the signal that drives the LPC adaptation. In either case, the purpose of the adaptation is to make the LPC model better fit the signal and thereby reduce the noise energy in the system.

In a subband coder, the obligatory blocks are the filter bank and a set of quantizers that operate on the outputs of the filter bank. Its block diagram is shown in Figure 3. A subband coder separates different frequencies into different bands by using a filter bank and then quantizes sets of bands differently. As the filter bank maintains the same total signal energy but divides it into multiple bands, each of which must have less than the total energy, the quantizers required in each of the frequency bands will have fewer steps and therefore will require fewer bits.

In most subband coders, the quantization is signal dependent, and a rate control system examines the signal (or analyzed signal) in order to control the quantizers. This sort of system, which was originally based on the idea of rate distortion theory [1], is also the beginning of a perceptual system, because the quantizing noise is limited to each band and no longer spreads across the entire signal bandwidth. This is modestly different from the noise shaping involved in the LPC coder, because in that coder the noise spreads across the whole band but can be shaped. In perceptual terms, the two are very similar; however, mechanisms for controlling noise in the case of subband coders are well evolved, whereas mechanisms for noise shaping in LPC, in other than speech applications, are currently rudimentary.

Finally, we come to the perceptual coder. Here, we use the term perceptual coder to refer to a coding system that calculates an express measure of a masking or just noticeable difference (JND) curve and then uses that measure to control noise injection. Figure 4 shows the high-level block diagram for a standard perceptual coder. There are four parts to the perceptual coder: a filter bank, a perceptual model, the quantizer, and rate loop, and noiseless compressor–bitstream formatter.

The filter bank has the same function as the filter bank in the subband coder; i.e., it breaks the signal up into a time-frequency tiling. In the perceptual coder, the filter bank is most often switched; i.e., it has two or more different time-frequency tilings and a way to switch seamlessly between them. In MPEG-4, the filter banks are all modified discrete cosine transform (MDCT) [9] based.

The perceptual model is the heart of the perceptual coder. It takes the input signal,
sometimes the filtered signal, and other information from the rate loop and the coder setup and creates either a masking threshold or a set of signal-to-noise ratios that must be met during coding. There are many versions of such a model, such as in Brandenburg and Stoll [10], as well as many variations on how to carry out the process of modeling; however, they all do something that attempts to partition frequency, and sometimes time, into something approximating the frequency, and sometimes time, resolution of the cochlea.

The rate loop takes the filter bank output and the perceptual model, arranges the quantizers so that the bit rate is met, and also attempts to satisfy the perceptual criteria. If the perceptual criteria are left unmet, the rate loop attempts to do so in a perceptually inobtrusive fashion. In most such coders, a rate loop is an iterative mechanism that attempts some kind of optimization, usually heuristic, of rate versus quality.

Finally, because the quantizers required for addressing perceptual constraints are not particularly good in the information theoretic sense, a back-end coding using entropy-coding methods is usually present in order to pack the quantizer outputs efficiently into the bitstream. This bitstream formatter may also add information for synchronization, external data, and other functions.

II. MPEG-2 AAC—ADVANCED AUDIO CODING

The ISO/IEC MPEG-2 Advanced Audio Coding (AAC) technology [11,12] delivers unsurpassed audio quality at rates at or below 64 kbps/channel. Because of its high performance and because it is the most recent of the MPEG-2 audio coding standards (effectively being developed in parallel with the MPEG-4 standard), it was incorporated directly into the MPEG-4 General Audio Standard. It has a very flexible bitstream syntax that supports multiple audio channels, subwoofer channels, embedded data channels, and multiple programs consisting of multiple audio, subwoofer, and embedded data channels. Strike AAC combines the coding efficiencies of a high-resolution filter bank, backward-adaptive prediction, joint channel coding, and Huffman coding with a flexible coding architecture to permit application-specific functionality while still delivering excellent signal compression.

AAC supports a wide range of sampling frequencies (from 8 to 96 kHz) and a wide range of bit rates. This permits it to support applications ranging from professional or home theater sound systems through Internet music broadcast systems to low (speech) rate speech and music preview systems.

A block diagram of the AAC encoder is shown in Figure 5. The blocks are as follows:

*Filter bank:* AAC uses a resolution-switching filter bank that can switch between a high-frequency-resolution mode of 1024 (for maximum statistical gain during intervals of signal stationarity) and a high-time-resolution mode of 128 bands (for maximum time-domain coding error control during intervals of signal nonstationarity).

*TNS:* The Temporal Noise Shaping (TNS) tool modifies the filter bank characteristics so that the combination of the two tools is better able to adapt to the time-frequency characteristics of the input signal [13].

*Perceptual model:* A model of the human auditory system that sets the quantization noise levels based on the loudness characteristics of the input signal.

*Intensity and coupling, mid/side (M/S):* These two blocks actually comprise three
tools, all of which seek to protect the stereo or multichannel signal from noise imaging while achieving coding gain based on correlation between two or more channels of the input signal [14–16].

**Prediction**: A backward adaptive recursive prediction that removes additional redundancy from individual filter bank outputs [17].

**Scale factors**: Scale factors set the effective step sizes for the nonuniform quantizers.

**Quantization, noiseless coding**: These two tools work together. The first quantizes the spectral components and the second applies Huffman coding to vectors of quantized coefficients in order to extract additional redundancy from the nonuniform probability of the quantizer output levels. In any perceptual encoder, it is very difficult to control the noise level accurately while at the same time achieving an "optimum" quantizer. It is, however, quite efficient to allow the quantizer to operate unconstrained and then to remove the redundancy in the probability density function (PDF) of the quantizer outputs through the use of entropy coding.

**Rate–distortion control**: This tool adjusts the scale factors such that more (or less) noise is permitted in the quantized representation of the signal, which, in turn, requires fewer (or more) bits. Using this mechanism, the rate–distortion control tool can adjust the number of bits used to code each audio frame and hence adjust the overall bit rate of the coder.

**Bitstream multiplexer**: The multiplexer assembles the various tokens to form a bitstream.

This section will discuss the blocks that contribute the most to AAC performance: the filter bank, the perceptual model, and noiseless coding.

### A. Analysis–Synthesis Filter Bank

The most significant aspect of the AAC filter bank is that it has high frequency resolution (1024 frequency coefficients) so that it is able to extract the maximum signal redundancy (i.e., provide maximum prediction gain) for stationary signals [1]. High frequency resolution also permits the encoder's psychoacoustic model to separate signal components that
differ in frequency by more than one critical band and hence extract the maximum signal irrelevance.

The AAC analysis-synthesis filter bank has three other characteristics that are commonly employed in audio coding: critical sampling, overlap-add synthesis, and perfect reconstruction. In a critically sampled filter bank, the number of time samples input to the analysis filter per second equals the number of frequency coefficients generated per second. This minimizes the number of frequency coefficients that must be quantized and transmitted in the bitstream. Overlap-add reconstruction reduces artifacts caused by block-to-block variations in signal quantization. Perfect reconstruction implies that in the absence of frequency coefficient quantization, the synthesis filter output will be identical, within numerical error, to the analysis filter input.

When transient signals must be coded, the high-resolution or "long-block" filter bank is not an advantage. For this reason, the AAC filter bank can switch from high-frequency-resolution mode to high-time-resolution mode. The latter mode, or "short-block" mode, permits the coder to control the anticausal spread of coding noise [18]. The top panel in Figure 6 shows the window sequence for the high-frequency-resolution mode, and the bottom panel shows the window sequence for a transition from long-block to short-block and back to long-block mode.

The filter bank adopted for use in AAC is a modulated, overlapped filter bank called the modified discrete cosine transform (MDCT) [9]. The input sequence is windowed (as shown in Fig. 6) and the MDCT computed. Because it is a critically sampled filter bank, advancing the window to cover 1024 new time samples produces 1024 filter bank output samples. On synthesis, 1024 filter bank input samples produce 2048 output time samples.

![Stationary Signal](image)

![Transient Signal](image)

**Figure 6** Window sequence during stationary and transient signal conditions.
which are then overlapped 50% with the previous filter bank result and added to form the output block.

B. Perceptual Model

The perceptual model estimates the threshold of masking, which is the level of noise that is subjectively just noticeable given the current input signal. Because models of auditory masking are primarily based on frequency domain measurements [7,19], these calculations are typically based on the short-term power spectrum of the input signal, and threshold values are adapted to the time–frequency resolution of the filter bank outputs. The threshold of masking is calculated relative to each frequency coefficient for each audio channel for each frame of input signal, so that it is signal dependent in both time and frequency. When the high-time-resolution filter bank is used, it is calculated for the spectra associated with each of the sequence of eight windows used in the time–frequency analysis. In intervals in which pre-echo distortion is likely, more than one frame of signal is considered such that the threshold in frames just prior to a nonstationary event is depressed to ensure that leakage of coding noise is minimized. Within a single frame, calculations are done with a granularity of approximately 1/3 Bark, following the critical band model in psychoacoustics. The model calculations are similar to those in psychoacoustic model II in the MPEG-1 audio standard [10].

The following steps are used to calculate the monophonic masking threshold of an input signal:

- Calculate the power spectrum of the signal in 1/3 critical band partitions.
- Calculate the tonelike or noiselike nature of the signal in those partitions, called the tonality measure.
- Calculate the spread of masking energy, based on the tonality measure and the power spectrum.
- Calculate time domain effects on the masking energy in each partition.
- Relate the masking energy to the filter bank outputs.

Once the masking threshold is known, it is used to set the scale factor values in each scale factor band such that the resulting quantizer noise power in each band is below the masking threshold in that band. When coding audio channel pairs that have a stereo presentation, binaural masking level depression must be considered [14].

C. Quantization and Noiseless Coding

The spectral coefficients are coded using one quantizer per scale factor band, which is a fixed division of the spectrum. For high-resolution blocks there are 49 scale factor bands, which are approximately 1/2 Bark in width. The psychoacoustic model specifies the quantizer step size (inverse of scale factor) per scale factor band. An AAC encoder is an instantaneously variable rate coder, but if the coded audio is to be transmitted over a constant rate channel, then the rate–distortion module adjusts the step sizes and number of quantization levels so that a constant rate is achieved.
1. **Quantization**

AAC uses a nonlinear quantizer for spectral component $x_i$ to produce $\hat{x}_i$:

$$\hat{x}_i = \text{sign}(x_i)n \text{ int}\left(\left(\frac{|x_i|}{\frac{\sqrt{2}}{\text{stepsize}}}\right)^{4/3}\right)$$ (1)

The main advantage of the nonlinear quantizer is that it shapes the noise as a function of the amplitude of the coefficients, such that the increase of the signal-to-noise ratio with increasing signal energy is much lower than that of a linear quantizer. The exponent step-size is the quantized step size in a given scale factor band. The first scale factor is PCM coded, and subsequent ones are Huffman coded differential values.

2. **Rate-Distortion Control**

The quantized coefficients are Huffman coded. A highly flexible coding method allows several Huffman tables to be used for one spectrum. Two- and four-dimensional tables with and without sign are available. The noiseless coding process is described in detail in Quackenbush and Johnston [20]. To calculate the number of bits needed to code a spectrum of quantized data, the coding process has to be performed and the number of bits needed for the spectral data and the side information has to be accumulated.

3. **Noiseless Coding**

The input to the noiseless coding is the set of 1024 quantized spectral coefficients and their associated scale factors. If the high-time-resolution filter bank is selected, then the 1024 coefficients are actually a matrix of 8 by 128 coefficients representing the time–frequency evolution of the signal over the duration of the eight short-time spectra. In AAC an extended Huffman code is used to represent $n$-tuples of quantized coefficients, with the Huffman code words drawn from one of 11 codebooks. The maximum absolute value of the quantized coefficients that can be represented by each Huffman codebook and the number of coefficients in each $n$-tuple for each codebook are shown in Table 1. There are two codebooks for each maximum absolute value, with each represent-

<table>
<thead>
<tr>
<th>Codebook index</th>
<th>Tuple size</th>
<th>Maximum absolute value</th>
<th>Signed values</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>4</td>
<td>0</td>
<td>Yes</td>
</tr>
<tr>
<td>1</td>
<td>4</td>
<td>1</td>
<td>Yes</td>
</tr>
<tr>
<td>2</td>
<td>4</td>
<td>2</td>
<td>No</td>
</tr>
<tr>
<td>3</td>
<td>4</td>
<td>2</td>
<td>No</td>
</tr>
<tr>
<td>4</td>
<td>2</td>
<td>4</td>
<td>Yes</td>
</tr>
<tr>
<td>5</td>
<td>2</td>
<td>4</td>
<td>Yes</td>
</tr>
<tr>
<td>6</td>
<td>2</td>
<td>7</td>
<td>No</td>
</tr>
<tr>
<td>7</td>
<td>2</td>
<td>7</td>
<td>No</td>
</tr>
<tr>
<td>8</td>
<td>2</td>
<td>12</td>
<td>No</td>
</tr>
<tr>
<td>9</td>
<td>2</td>
<td>12</td>
<td>No</td>
</tr>
<tr>
<td>10</td>
<td>2</td>
<td>16 (ESC)</td>
<td>No</td>
</tr>
</tbody>
</table>
ing a distinct probability distribution function. Codebooks can represent signed or unsigned values, and for the latter the sign bit of each nonzero coefficient is appended to the codeword.

Two codebooks require special note: codebook 0 and codebook 11. Codebook 0 indicates that all coefficients within a section are zero, requiring no transmission of the spectral values and scale factors. Codebook 11 can represent quantized coefficients that have an absolute value greater than or equal to 16 by means of an escape (ESC) Human code word and an escape code that follows it in the bitstream.

III. MPEG-4 ADDITIONS TO AAC

An important aspect of the overall MPEG-4 Audio [21] functionality is covered by the so-called General Audio (GA) part, i.e., the coding of arbitrary audio signals. MPEG-4 General Audio coding is built around the coder kernel provided by MPEG-2 Advanced Audio Coding (AAC) [22], which is extended by additional coding tools and coder configurations. The perceptual noise substitution (PNS) tool and the Long-Term Prediction (LTP) tool are available to enhance the coding performance for the noiselnike and very tonal signals, respectively. A special coder kernel (Twin VQ) is provided to cover extremely low bit rates. Together with a few additional tools and the MPEG-4 narrowband CELP coder, a flexible bit rate scalable coding system is defined including a variety of possible coder configurations. The following sections will describe these features in more detail.

A. Perceptual Noise Substitution

Generic audio coding predominantly employs methods for waveform coding, but other coding methods are conceivable that aim not at preserving the waveform of the input signal but at reproducing a perceptually equivalent output signal at the decoder end. In fact, relaxing the requirement for waveform preservation may enable significant savings in bit rate when parts of the signal are reconstructed from a compact parametric representation of signal features. The PNS tool [23] allows a very compact representation of noiselnike signal components and in this way further increases compression efficiency for certain types of input signals.

The PNS technique is based on the observation that the perception of noiselnike signals is similar regardless of the actual waveform of the stimulus provided that both the spectral envelope and the temporal fine structure of the stimuli are similar.

The PNS tool exploits this phenomenon in the context of the MPEG-4 perceptual audio coder within a coder framework based on analysis–synthesis filter banks. A similar system was proposed and investigated by Schulz [24,25]. The concept of the PNS technique can be described as follows (see Fig. 7):

In the encoder, noiselnike components of the input signal are detected on a scale factor band basis.

The groups of spectral coefficients belonging to scale factor bands containing noiselnike signal components are not quantized and coded as usual but omitted from the quantization–coding process.

Instead, only a noise substitution flag and the total power of the substituted spectral coefficients are transmitted for each of these bands.
Review of MPEG–4 General Audio Coding

![Diagram of MPEG–4 General Audio Coding](image)

Figure 7  The principle of perceptual noise substitution.

In the decoder, pseudorandom vectors with the desired total noise power are inserted for the substituted spectral coefficients.

This approach will result in a highly compact representation of the noiselike spectral components because only the signaling and the energy information is transmitted per scale factor band rather than codebook, scale factor, and the set of quantized and coded spectral coefficients.

The PNS tool is tightly integrated into the MPEG–4 coder framework and reuses many of its basic coding mechanisms. As a result, the additional decoder complexity associated with the PNS coding tool is very low in terms of both computational and memory requirements. Furthermore, because of the means of signaling PNS, the extended bitstream syntax is downward compatible with MPEG–2 AAC syntax in the sense that each MPEG–2 AAC decoder will be able to decode the extended bitstream format as long as the PNS feature is not used.

B. Long-Term Prediction

Long-term prediction is a technique that is well known from speech coding and has been used to exploit redundancy in the speech signal that is related to the signal periodicity as manifested by the speech pitch (i.e., pitch prediction). Whereas common speech coders apply long-term prediction within the framework of a time domain coder, the MPEG–4 Audio LTP tool has been integrated into the framework of a generic perceptual audio coder; i.e., quantization and coding are performed on a spectral representation of the input signal. Figure 8 shows the combined LTP–coding system.

As shown in the figure, the LTP is used to predict the input signal based on the quantized values of the preceding frames, which were transformed back to a time domain representation by the inverse (synthesis) filter bank and the associated inverse TNS opera-
Figure 8 The LTP in the MPEG-4 GA coder.

By comparing this decoded signal with the input signal, the optimal pitch lag and gain factor are determined. In the next step, both the input signal and the predicted signal are mapped to a spectral representation via an analysis filter bank and a forward TNS operation. Depending on which alternative is more favorable, coding of either the difference (residual) signal or the original signal is selected on a scale factor band basis. This is achieved by means of a so-called frequency-selective switch (FSS), which is also used in the context of the MPEG-4 GA scalable systems (see Sec. I.C.5). Because of the underlying principle, the LTP tool provides optimal coding gain for stationary harmonic signals (e.g., 'pitch pipe') as well as some gain for nonharmonic tonal signals (e.g., polyphonic tonal instruments). Compared with the rather complex MPEG-2 AAC predictor tool, the LTP tool shows a saving of approximately one-half in both computational complexity and memory requirements.

C. Twin VQ

The Transform-Domain Weighted Interleaved Vector Quantization (Twin VQ) [26,27] is an alternative VQ-based coding kernel that is designed to provide good coding performance at extremely low bit rates (at or below 16 kbit/sec). It is used in the context of the MPEG-4 scalable GA system (see Sec. I.C.5). The coder kernel is adapted to operate within the spectral representation provided by the AAC coder filter bank [28].

The Twin VQ kernel performs a quantization of the spectral coefficients in two steps. In the first step the spectral coefficients are normalized to a specified target range and are then quantized by means of a weighted vector quantization process. The spectral normalization process includes a linear predictive coding (LPC) spectral estimation scheme, a periodic component extraction scheme, a Bark-scale spectral estimation scheme, and a power estimation scheme, which are carried out sequentially. As a result, the spectral
coefficients are “flattened” and normalized across the frequency axis. The parameters associated with the spectral normalization process are quantized and transmitted as side information. In the second step, called the weighted vector quantization process, the attenuated spectral coefficients are interleaved and divided into subvectors for vector quantization. For each subvector, a weighted distortion measure is applied to the conjugate structure VQ, which uses a pair of codebooks. In this way, perceptual control of the quantization distortion is achieved. The main part of the transmitted information consists of the selected codebook indices. Because of the nature of the interleaved vector quantization scheme, no adaptive bit allocation is carried out for individual quantization indices (an equal amount of bits is spent for each of the quantization indices).

The spectral normalization process includes the following steps:

**LPC spectral estimation:** At the first stage of the spectrum normalization process, the overall spectral envelope is estimated by means of an LPC model and used to normalize the spectral coefficients. This allows efficient coding of the envelope using line spectral pair (LSP) parameters.

**Periodic component coding:** If the frame is coded using one long filter bank window, “periodic peak” components are coded. This is done by estimating the fundamental signal period (pitch) and extracting a number of periodic peak components from the flattened spectral coefficients. The data are quantized together with the average gain of these components.

**Bark-scale envelope coding:** The resulting coefficients are further flattened by using a spectral envelope based on the Bark-related AAC scale factor bands. The envelope values are normalized and quantized by means of a vector quantizer with inter-frame prediction.

The weighted VQ process comprises the following steps:

**Interleaving of spectral coefficients:** Prior to vector quantization, the attenuated spectral coefficients are interleaved and divided into subvectors as shown in Figure 9. If the subvectors were constructed from spectral coefficients that were consecutive in frequency, the subvectors corresponding to the lower frequency range would require much finer quantization (more bits) than those corresponding

*Figure 9* Twin VQ spectral coefficient interleaving.
to higher frequencies. In contrast, interleaving allows more constant bit allocation for each subvector. Perceptual shaping of the quantization noise can be achieved by applying an adaptive weighted distortion measure that is associated with the spectral envelope and the perceptual model.

*Vector quantization:* The vector quantization part uses a two-channel conjugate structure with two sets of codebooks. The best combination of indices is selected to minimize the distortion when two code vectors are added. This approach decreases both the amount of memory required for the codebooks and the computational demands for the codebook search.

The Twin VQ coder kernel operates at bit rates of 6 kbit/sec and above and is used mainly in the scalable configurations of the MPEG-4 GA coder.

**D. MPEG-4 Scalable General Audio Coding**

Today's popular schemes for perceptual coding of audio signals specify the bit rate of the compressed representation (bitstream) during the encoding phase. Contrary to this, the concept of scalable audio coding enables the transmission and decoding of the bitstream with a bit rate that can be adapted to dynamically varying requirements, such as the instantaneous transmission channel capacity. This capability offers significant advantages for transmitting content over channels with a variable channel capacity (e.g., the Internet, wireless transmission) or connections for which the available channel capacity is unknown at the time of encoding.

To achieve this, bitstreams generated by scalable coding schemes consist of several partial bitstreams that can be decoded on their own in a meaningful way. In this manner, transmission (and decoding) of a subset of the total bitstream will result in a valid, decodable signal at a lower bit rate and quality. Because bit-rate scalability is considered one of the core functionalities of the MPEG-4 standard, a number of scalable coder configurations are described by the standard. In the context of MPEG-4 GA coding, the key concept of scalable coding can be described as follows (see Fig. 10):

The input signal is coded–decoded by a first coder (coder 1, the *base layer coder*) and the resulting bitstream information is transmitted as a first part of the composite scalable bitstream.

Next, the coding error signal is calculated as the difference between the encoded–

![Figure 10](image-url)  Basic concept of MPEG-4 scalable GA coding.
decoded signal and the original signal. This signal is used as the input signal of the next coding stage (coder 2), which contributes the next part of the composite scalable bitstream.

This process can be continued as often as desired (although, practically, no more than three or four coders are used). While the first coding stage (usually called base layer) transmits the most relevant components of the signal at a basic quality level, the following stages subsequently enhance the coding precision delivered by the preceding layers and are therefore called enhancement layers.

E. MPEG–4 Scalable Audio Coder

Figure 11 shows the structure of an MPEG–4 scalable audio coder. In this configuration, the base layer coder (called core coder) operates at a lower sampling frequency than the enhancement layer coder, which is based on AAC [29]:

The input signal is downsampled and encoded by the core coder. The resulting core layer bitstream is both passed on to the bitstream multiplexer and decoded by a local core decoder. The decoded output signal is upsampled to the rate of the enhancement layer encoder and passed through the MDCT analysis filter bank.

In a parallel signal path, the delay-compensated input signal is passed through the MDCT analysis filter bank.

The frequency-selective switch (FSS) permits selection between coding of spectral coefficients of the input signal and coding of spectral coefficients of the difference (residual) signal on a scale factor band basis. The assembled spectrum is passed to the AAC coding kernel for quantization and coding. This results in an enhancement layer bitstream that is multiplexed into the composite output bitstream.

Significant structural simplifications to this general scheme are achieved if both the base layer and the enhancement layer coders are filter bank–based schemes (i.e., AAC or Twin VQ). In this case, all quantization and coding are carried out on a common set of spectral coefficients and no sampling rate conversion is necessary [23]. The structure of a core-based scalable decoder is shown in Figure 12: The composite bitstream is first demultiplexed into base layer and enhancement layer bitstreams.

Then the core layer bitstream is decoded, upsampled, delay compensated, and passed into the IMDCT synthesis filter bank. If only the core layer bitstream is received in a decoder, the output of the core layer decoder is presented via an optional postfilter.

If higher layer bitstreams are also available to the decoder, the spectral data are

![Diagram](image)

**Figure 11** Structure of an MPEG–4 scalable coder.
decoded from these layers and accumulated over all enhancement layers. The resulting spectral data are combined with the spectral data of the core layer as controlled by the FSS and transformed back into a time domain representation.

Within the MPEG–4 GA scalable coding system, certain restrictions apply regarding the order and role of various coder types:

- The MPEG–4 narrowband CELP coder 1.4.3 can be used as a core coder.
- The Twin VQ coder kernel can act as a base layer coder or as an enhancement layer coder if coding of the previous layer is based on Twin VQ as well.
- The AAC coder can act as both a base layer and an enhancement layer coder.

One interesting configuration is the combination of a CELP core coder and several AAC-based enhancement layers, which provides very good speech quality even at the lowest layer decoded output.

F. Mono–Stereo Scalability

Beyond the type of scalability described up to now, the MPEG–4 scalable GA coder also provides provisions for mono–stereo scalability. Decoding of lower layers results in a mono signal, whereas decoding of higher layers will deliver a stereo signal after decoding [30]. This useful functionality is achieved in the following way:

All mono layers operate on a mono version of the stereo input signal. The stereo enhancement layers encode the stereo signal as either an M/S (mid/side) or L/R (left/right) representation, as known from AAC joint stereo coding. When using an M/S representation, the encoded signal from the lower mono layers is available as an approximation of the mid signal.

IV. THE REST OF THE MPEG–4 NATURAL AUDIO CODER

A. Target Applications

The MPEG–4 general audio coder is the first that covers the whole range of low-bit-rate audio coding applications. MPEG–4 Audio has enough capabilities to replace virtually all of the existing audio coding standards and offers many additional functionalities not
available with any other coder, such as bit rate scalability. It might be asked why such
an all-in-one system is needed and whether it might be oversized for many applications;
in reality, such a system is needed for upcoming communication networks. It seems very
likely that telephone and computer networks will soon merge into a unified service. With
the start of music and video distribution over computer networks, traditional broadcasting
services will start to merge into this global communication web. In such an environment
the channel to a specific end user can be anything from a low-bit-rate cellular phone
connection to a gigabit computer network connected operating over fiber-optic lines. With-
out a universal coding system, transcoding is frequently required, e.g., when a cell phone
user communicates with an end user via a future hi-fi set that normally retrieves high-
quality 11-channel music material from the Internet.

B. General Characteristics

As with MPEG–1 Audio and MPEG–2 Audio, the low-bit-rate coding of audio signals
is the core functionality in MPEG–4. The lower end of this bit rate range is marked by
pure speech coding techniques, starting from 2 kbit/sec. The upper end reaches up to more
than a 100 kbit/sec per audio channel for transparent coding of high-quality audio material,
with a dynamic range, sampling frequency options, and a multichannel audio capability
that exceed the standard set by today’s compact disc. For example, seven-channel stereo
material, at a sampling rate of 96 kHz, can be encoded with a dynamic range of more
than a 150 dB if desired.

This broad range cannot be covered by a single coding scheme but requires the
combination of several algorithms and tools.* These are integrated into a common frame-
work with the possibility of a layered coding, starting with a very low bit rate speech
coder and additional general audio coding layers on top of this base layer. Each layer
further enhances the audio quality available with the previous layer. With such a scheme
at any point in the transmission chain the audio bitstream can be adapted to the available
bit rate by simply dropping enhancement layers.

In general, there are two types of coding schemes in MPEG–4 Audio. Higher bit
rate applications are covered by the MPEG–4 General Audio (GA) coder 1.3, which in
general is the best option for bit rates of 16 kbit/sec per channel and above for all types
of audio material. If lower rates are desired, the GA coder, with a lower bit rate limit of
around 6 kbit/sec, is still the best choice for general audio material. However, for speech-
dominated applications the MPEG–4 speech coder is available as an alternative, offering
bit rates down to 2 kbit/sec.

C. The MPEG–4 Speech Coder

1. Introduction

The speech coder in MPEG–4 Audio transfers the universal approach of the “traditional”
MPEG audio coding algorithms to the speech coding world. Whereas other speech coder
standards, e.g., G.722 [31], G.723.1 [32], or G.729 [33], are defined for one specific sam-

* A tool in MPEG–4 Audio is a special coding module that can be used as a component in different
coding algorithms.
pling rate and for one to at most three different bit rates, the MPEG-4 coder supports
multiple sampling rates and more than 50 different bit rate options. Furthermore, embed-
ded coding techniques are available that allow decoding subsets of the bitstream into valid
output signals. The following section gives an overview of the functionalities of the
MPEG-4 speech coder.

2. Speech Coder Functionalities

   a. Narrowband Speech Coding. The term narrowband speech coding usually
describes the coding of speech signals with an audio bandwidth of around 3.5 kHz. The
MPEG-4 coder, like other digital narrowband speech coders, supports this with a sampling
rate of 8 kHz. In addition, primarily to support scalable combinations with AAC enhance-
ment layers (see Sec. I.C.5), the MPEG-4 narrowband coder allows slightly different
sampling rates close to 8 kHz, e.g., 44,100/6 = 7350 Hz. The available bit rates range
from 2 kbit/sec* up to 12.2 kbit/sec. The algorithmic delay† ranges from around 40 msec
for the lowest bit rates to 25 msec at around 6 kbit/sec and down to 15 msec for the
highest rates. A trade-off between audio quality and delay is possible, as some bit rates
are available with a different coding delay All but the lowest bit rate are realized with a
coder based on CELP coding techniques. The 2 kbit/sec coder is based on the novel
HVXC (see Sec. I.D.3) scheme.

   b. Wideband Speech Coding. Wideband speech coding, with a sampling rate of
16 kHz, is available for bit rates from 10.9 to 21.1 kbit/sec with an algorithmic delay of
25 msec and for bit rates from 13.6 to 23.8 kbit/sec with a delay of 15 msec. The wideband
coder is an upscaled narrowband CELP coder, using the same coding tools, but with
different parameters.

   c. Bit Rate Scalability. Bit rate scalability is a unique feature of the MPEG-4
Audio coder set, which is used to realize a coding system with embedded layers. A nar-
rowband or a wideband coder, as just described, is used as a base layer coder. On top of
that, additional speech coding layers can be added which, step by step, increase the audio
quality.‡ Decoding is always possible if at least the base layer is available. Each enhance-
ment layer for the narrowband coder uses 2 kbit/sec. The step size for the wideband coder
is 4 kbit/sec. Although only one enhancement layer is possible for HVXC, up to three
bit rate scalable enhancement layers (BRSELS) may be used for the CELP coder.

   d. Bandwidth Scalability. Bandwidth scalability is another option to improve the
audio quality by adding an additional coding layer. Only one bandwidth scalable enhance-
ment layer (BWSEL) is possible and can be used in combination with the CELP coding
tools but not with HVXC. In this configuration, a narrowband CELP coder at a sampling
rate of 8 kHz first codes a downsampled version of the input signal. The enhancement
layer, running with a sampling rate of 16 kHz, expands the audio bandwidth from 3.5 to
7 kHz.

3. The Technology

All variants of the MPEG-4 speech coder are based on a model using an LPC filter [34]
and an excitation module [35]. There are three different configurations, which are listed

* A variable rate mode with average bit rates below 2 kbit/sec is also available.
† The algorithmic delay is the shortest possible delay, assuming zero processing and transmission
time.
‡ For a narrowband base coder, GA enhancement layers, as well as speech layers, are possible.
in Table 2. All of these configurations share the same basic method for the coding of the LPC filter coefficients. However, they differ in the way the filter excitation signal is transmitted.

4. Coding of the LPC Filter Coefficients

In all configurations, the LPC filter coefficients are quantized and coded in the LSP domain [36,37]. The basic block of the LSP quantizer is a two-stage split-VQ design [38] for 10 LSP coefficients, shown in Figure 13. The first stage contains a VQ that codes the 10 LSP coefficients either with a single codebook (HVXC, CELP BWSEL) or with a split VQ with two codebooks for 5 LSP coefficients each. In the second stage, the quantization accuracy is improved by adding the output of a split VQ with two tables for $2 \times 5$ LSP coefficients. Optionally, inter-frame prediction is available in stage two, which can be switched on and off, depending on the characteristics of the input signal. The codebook in stage two contains two different sets of vectors to be used, depending on whether or not the prediction is enabled.

The LSP block is applied in six different configurations, which are shown in Table 3. Each configuration uses its own set of specifically optimized VQ tables. At a sampling rate of 8 kHz a single block is used. However, different table sets, optimized for HVXC and the narrowband CELP coder, are available. At 16 kHz, there are 20 LSP coefficients and the same scheme is applied twice using two more table sets for the independent coding of the lower and upper halves of the 20 coefficients.

Although so far this is conventional technology, a novel approach is included to support the bandwidth scalability option of the MPEG-4 CELP coder [39] (Fig. 14). The

![Figure 13](image-url) **Figure 13** Two-stage split-VQ inverse LSP quantizer with optional inter-frame predictor.
### Table 3  LSP-VQ Table Sizes for the Six LSP Configurations (Entries × Vector Length)

<table>
<thead>
<tr>
<th>LSP configuration</th>
<th>Table size stage 1</th>
<th>Table size stage 2 (with prediction)</th>
<th>Table size stage 2 (without prediction)</th>
</tr>
</thead>
<tbody>
<tr>
<td>HVXC 8 kHz</td>
<td>32 × 10</td>
<td>64 × 5 + 16 × 5</td>
<td>64 × 5 + 16 × 5</td>
</tr>
<tr>
<td>CELP 8 kHz</td>
<td>16 × 5 + 16 × 5</td>
<td>64 × 5 + 32 × 5</td>
<td>64 × 5 + 32 × 5</td>
</tr>
<tr>
<td>CELP 16 kHz (lower)</td>
<td>32 × 5 + 32 × 5</td>
<td>64 × 5 + 64 × 5</td>
<td>64 × 5 + 64 × 5</td>
</tr>
<tr>
<td>CELP 16 kHz (upper)</td>
<td>16 × 5 + 16 × 5</td>
<td>64 × 5 + 16 × 5</td>
<td>64 × 5 + 16 × 5</td>
</tr>
<tr>
<td>CELP BWSEL (lower)</td>
<td>16 × 10</td>
<td>-</td>
<td>16 × 5 + 64 × 5</td>
</tr>
<tr>
<td>CELP BWSEL (upper)</td>
<td>128 × 10</td>
<td>-</td>
<td>128 × 5 + 16 × 5</td>
</tr>
</tbody>
</table>

![Network Diagram](image)

**Figure 14** Bandwidth scalable LSP inverse quantization scheme.

Quantized LSP coefficients of the first (narrowband) layer are reconstructed and transformed to the 16-kHz domain to form a first-layer coding of the lower half of the wideband LSP coefficients. Two additional LSP-VQ blocks provide a refinement of the quantization of the lower half of the LSP coefficients and code the upper half of the coefficients. Again, a different set of optimized VQ tables is used for this purpose.*

### 5. Excitation Coding

For the coding of the LPC coefficients, one common scheme is used for all configurations, bit rates, and sampling rates. However, several alternative excitation modules are required to support all functionalities:

1. **MPE**: The broadest range of bit rates and functionalities is covered by a multimode multipulse excitation (MPE) [3,39,40] tool, which supports narrowband wideband coding, as well as bit rate and bandwidth scalability.

2. **RPE**: Because of the relatively high complexity of the MPEG-4 MPE tool if used for a wideband encoder, a regular pulse excitation (RPE) [41] module is...

---

*For completeness: The predictor of the BWSEL specific LSP decoder blocks is applied not to the second stage only but rather to both stages together.
available as an alternative, low-complexity encoding option, resulting in slightly lower speech quality [42].

3. **HVXC**: To achieve good speech quality at very low rates, an approach quite different from MPE or RPE is required. This technique is called harmonic vector excitation coding (HVXC) [43]. It achieves excellent speech quality even at a bit rate of only 2 kbit/sec in the MPEG–4 speech coder verification tests [42] for speech signals with and without background noise. HVXC uses a completely parametric description of the coded speech signal and is therefore also called the MPEG–4 parametric speech coder. Whereas the CELP coding modes offer some limited performance (compared with the MPEG–4 GA coder) for nonspeech signals, HVXC is recommended for speech only because its model is designed for speech signals.

a. **Multipulse Excitation (MPE)**. The maximum configuration of the MPE excitation tool, comprising the base layer, all three bit rate scalable layers (BRSELs), and the bandwidth scalable BWSEL, is shown in Figure 15.

The base layer follows a layout that is often found in a CELP coder. An adaptive codebook [44] which generates one component of the excitation signal, is used to remove the redundancy of periodic signals. The nonperiodic content is represented with a multipulse signal. An algebraic codebook [45] structure is used to reduce the side information that is required to transmit the locations and amplitudes of the pulses. The base layer operates at a sampling rate of either 8 or 16 kHz. On top of this base layer, two types of

![Figure 15](image-url) Block diagram of the maximum configuration of the MPE excitation tool, including bit rate (BRSEL) and bandwidth scalability (BWSEL) extension layers.
enhancement layers are possible. The BRSEL-type layers add additional excitation pulses with an independent gain factor. The combined outputs of the base layer and any number of BRSELS always share the sampling frequency of the base layer. If a BWSEL is added, with or without an arbitrary number of BRSELS, however, the excitation signal produced by the BWSEL output always represents a 16-kHz signal. In this case, the base layer and the BRSEL are restricted to a sampling rate of 8 kHz.

b. Regular Pulse Excitation (RPE). Although the MPEG-4 MPE module supports all the functionalities of the RPE tool with at least the same audio quality [42], RPE was retained in the MPEG-4 speech coder tool set to provide a low-complexity encoding option for wideband speech signals. Whereas the MPEG-4 MPE tool uses various VQ techniques to achieve optimal quality, the RPE tool directly codes the pulse amplitudes and positions. The computational complexity of a wideband encoder using RPE is estimated to be about one-half that of a wideband encoder using MPE. All MPEG-4 Audio speech decoders are required to support both RPE and MPE. The overhead for the RPE excitation decoder is minimal. Figure 16 shows the general layout of the RPE excitation tool.

c. HVXC. The HVXC decoder, shown in Figure 17, uses two independent LPC synthesis filters for voiced and unvoiced speech segments in order to avoid the voiced excitation being fed into the unvoiced synthesis filter and vice versa. Unvoiced excitation components are represented by the vectors of a stochastic codebook, as in conventional CELP coding techniques. The excitation for voiced signals is coded in the form of the spectral envelope of the excitation signal. It is generated from the envelope in the harmonic synthesizer using a fast IFFT synthesis algorithm. To achieve more natural speech quality, an additional noise component can be added to the voiced excitation signal. Another feature of the HVXC coder is the possibility of pitch and speech change, which is directly facilitated by the HVXC parameter set.

6. Postfilter

To enhance the subjective speech quality, a postfilter process is usually applied to the synthesized signal. No normative postfilter is included in the MPEG-4 specification, although examples are given in an informative annex. In the MPEG philosophy, the postfilter is the responsibility of the manufacturer as the filter is completely independent of the bitstream format.

![Figure 16](image-url)  
Figure 16  Block diagram of the RPE tool.
V. CONCLUSIONS

Based in part on MPEG–2 AAC, in part on conventional speech coding technology, and in part in new methods, the MPEG–4 General Audio coder provides a rich set of tools and features to deliver both enhanced coding performance and provisions for various types of scalability. The MPEG–4 GA coding defines the current state of the art in perceptual audio coding.

ACKNOWLEDGMENTS

The authors wish to express their thanks to Niels Rump, Lynda Feng, and Rainer Martin for their valuable assistance in proofreading and \LaTeX{} typesetting.

REFERENCES

10. K Brandenburg, G Stoll. ISO-MPEG-1 Audio: A generic standard for coding of high quality
digital audio. In: N Gilchrist, C Grewin, eds. Collected Papers on Digital Audio Bit-Rate Re-
pictures and audio. Part 7: Advanced audio coding.
12. M Bosi, K Brandenburg, S Quackenbush, L Fielder, K Akagiri, H Fuchs, M Diets, J Herre,
noise shaping (TNS). 101st AES Convention, Los Angeles, November 1996.
569–571.
16. JD Johnston, J Herre, M Davis, U Gbur. MPEG-2 NBC audio—stereo and multichannel coding
methods. Presented at the 101st AES Convention, Los Angeles, November 1996.
17. H Fuchs. Improving MPEG Audio coding by backward adaptive linear stereo prediction. Pre-
sented at the 99th AES Convention, New York, October 1995, preprint 4086 (J-1).
20. SR Quackenbush, JD Johnston. Noiseless coding of quantized spectral components in MPEG-
2 Advanced Audio Coding. IEEE Workshop on Applications of Signal Processing to Audio
diosignal objects: Audio.
ing of moving pictures and associated audio: Advanced Audio Coding.
23. J Herre, D Schulz. Extending the MPEG-4 AAC codec by perceptual noise substitution. 104th
1996.
25. D Schulz. Kompression qualitativ hochwertiger digitaler Audio signale durch Rauschextrak-
101st AES Convention, Los Angeles, 1996, preprint 4377.
28. J Herre, E Allamanche, K Brandenburg, M Dietz, B Teichmann, B Grill, A Jin, T Moriya,
N Iwakami, T Norimatsu, M Tsushima, T Ishikawa. The integrated filterbank based scalable
29. B Grill. A bit rate scalable perceptual coder for MPEG-4 Audio. 103rd AES Convention, New
30. B Grill, B Teichmann. Scalable joint stereo coding. 105th AES Convention, San Francisco,
32. ITU-T. Recommendation G.723.1: Dual rate speech coder for multi-media communications
transmitting at 5.3 and 6.3 kbit/s, 1996.