ResearchGate

See discussions, stats, and author profiles for this publication at: https://www.researchgate.net/publication/226641498

Multichannel Audio Coding for Multimedia Services in Intelligent Environments

Chapter *in* Studies in Computational Intelligence · June 2008

DOI: 10.1007/978-3-540-78502-6_5

CITATION
1

READS
2,072

2 authors:

Athanasios Mouchtaris
Amazon
180 PUBLICATIONS **2,455** CITATIONS
SEE PROFILE

Panagiotis Tsakalides
Foundation for Research and Technology Hellas
249 PUBLICATIONS **5,172** CITATIONS
SEE PROFILE

All content following this page was uploaded by Panagiotis Tsakalides on 23 May 2014.

The user has requested enhancement of the downloaded file.

Multichannel Audio Coding for Multimedia Services in Intelligent Environments

Athanasios Mouchtaris and Panagiotis Tsakalides

Foundation for Research and Technology - Hellas (FORTH), Institute of Computer Science (ICS), and University of Crete, Department of Computer Science

Abstract. Audio is an integral component of multimedia services in intelligent environments. Use of multiple channels in audio capturing and rendering offers the advantage of recreating arbitrary acoustic environments, immersing the listener into the acoustic scene. On the other hand, multichannel audio contains a large degree of information which is highly demanding to transmit, especially for real-time applications. For this reason, a variety of compression methods have been developed for multichannel audio content. In this chapter, we initially describe the currently popular methods for multichannel audio compression. Low-bitrate encoding methods for multichannel audio have also been recently starting to attract interest, mostly towards extending MP3 audio coding to multichannel audio recordings, and these methods are also examined here. For synthesizing a truly immersive intelligent audio environment, interactivity between the user(s) and the audio environment is essential. Towards this goal, we present recently proposed multichannel-audio-specific models, namely the source/filter and the sinusoidal models, which allow for flexible manipulation and high-quality low-bitrate encoding, tailored for applications such as remote mixing and distributed immersive performances.

1 Introduction

In this chapter, audio coding methods are described, the emphasis being on multichannel audio coding. Multichannel audio is an important component in most of today's multimedia applications including entertainment (Digital Television-DTV, home-theaters), consumer audio, teleconferencing, PC games, *etc.*

Multichannel audio places a number of loudspeakers around the listener, and different content for each loudspeaker must be available. The most popular format for home-theater systems is the 5.1 multichannel format, which places 5 loudspeakers around the listener according to the recommendation in [1], as shown in Fig. 1. The .1 channel corresponds to an additional loudspeaker for low-frequency sounds (Low Frequency Effects - LFE channel) up to 120 Hz (subwoofer), which can be placed anywhere around the listener since these sounds are non-directional (it is usually placed in front or at the sides of the listener). Other formats such as 6.1 or 7.1 have also been proposed.

Intelligent environments are technology-enhanced environments that respond and adapt to the user's needs in an unobtrusive manner. Multimedia services in

intelligent environments include the delivery, storage and presentation of multimedia to the user, and certainly multichannel audio is an important part of these services. Immersion of the user into a virtual environment is central to the concept of intelligent environments, and multichannel audio can offer a spatial dimension in today's multimedia without a significant cost to the average consumer. This chapter includes MPEG Layer III (MP3) audio coding and MPEG AAC (Advanced Audio Coding), which are some of the most popular technologies for the delivery and storage of audio. In fact, in recent years, these technologies have resulted in nothing less of a revolution in the way that music is delivered and stored, with the proliferation of the Internet playing a central role in this process. While other multichannel audio coding methods have also been proposed and become popular (*e.g.,* from Dolby and DTS), we only describe MPEG audio coding methods as representative, given the space limitations and because our main goal is to provide a conceptual-level description of algorithms in this domain. Instead, we chose to include in this chapter methods that are currently under development (such as MPEG Surround), which are concerned with low bitrate coding of multichannel audio signals. These methods have gained attention in recent years since, while broadcasting of multichannel audio in AAC format *e.g.,* for DTV is practically feasible, Internet streaming of multichannel audio still remains a challenge.

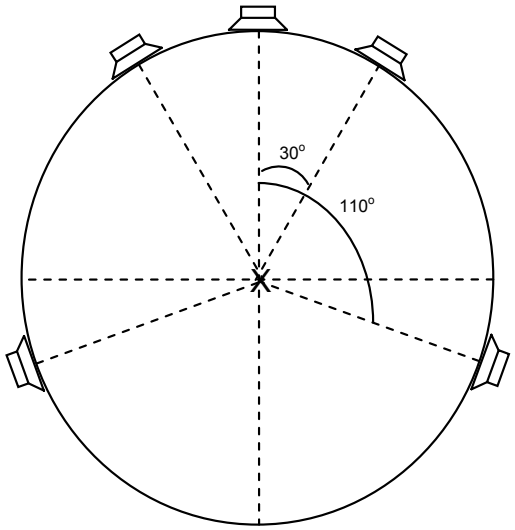

Fig. 1. ITU-R 5.1 multichannel audio reference configuration [1].

MPEG audio coding algorithms are part of a large family of methods known as subband coding. In general, most audio coding methods attempt to exploit the

human auditory system in order to shape the quantization noise so that it will be inaudible. Since the human auditory system has different tolerance for noise at different frequencies, audio coding methods are based on a subband or transform front-end, which transforms the signals in the frequency domain. Transform coding algorithms include methods where the audio signals are transformed in the frequency domain by a transform such as the DFT (Discrete Fourier Transform) or the DCT (Discrete Cosine Transform). Subband coders transform the audio signals in the frequency domain by dividing their spectral content with a bank of bandpass filters. Historically, subband coders resulted in lower frequency resolution than transform coders due to the limitation in the filter design. However, more recently high-resolution filterbanks with perfect reconstruction have been proposed (such as the Modified Discrete Cosine Transform - MDCT), and the distinction between subband and transform coding as terms has become artificial. These terms however can be used to distinguish them from parametric methods for audio coding, such as sinusoidal modeling and the source/filter model. For low bitrate coding, parametric methods are essential; these methods usually result in degradation of the audio signals when the available bandwidth is low. Current attempts for multichannel audio coding resulted in parametric coding of the binaural cues of multichannel audio signals with respect to a reference signal (downmix). These methods were found to result in high-quality low bitrate coding and are currently at the standardization stage (MPEG Surround). For low bitrates, the trade-off in this method is mostly about the reduced spatial image and not the audio quality. We should note that the term high audio quality corresponds to CD-like quality (also known as transparent coding), *i.e.,* when the coded waveform is perceptually indistinguishable from the original PCM audio signal.

Apart from MPEG Surround, other parametric methods, such as sinusoidal and source/filter models, have been found to degrade audio quality for low bitrates, for wideband audio signals, and have been mainly applied in speech coding. In this chapter, we describe recently proposed algorithms that adapt both the sinusoidal and source/filter models for high-quality low-bitrate audio coding. Again, these approaches achieve low datarates for a given spatial image, without sacrificing audio quality. Additionally, these methods are focused on coding the microphone signals of audio before mixing them into the final multichannel audio mix. This is an important advantage since it allows for remote mixing applications, offering interactivity between the user and his environment, which is of key importance for achieving truly immersive intelligent environments.

In the following sections, popular methods for stereo and multichannel audio coding are described. The accompanying list of references is representative and by no means exhaustive. The topics examined in this chapter aim at describing the most popular methods for audio coding not only at the present time, but also those that have been prominent in the past and those that are expected to dominate the field in the future. In this manner, we attempt to give the reader a better perspective of the challenges in the field of multichannel audio coding, and to provide a motivating historical background. In Section 2, we start by

describing MP3 audio coding, which dominated the area of stereophonic audio signal coding. In Section 3, we analyze MPEG-2 AAC coding, which essentially extends MP3 coding to multichannel audio, and is the algorithm behind most multimedia applications of today and the near future. In Section 4, we describe in more detail the principles in MP3 and AAC coding for exploiting interchannel similarities. These are examined in a separate section since most of multichannel audio coding research today is focused on this problem. In Section 5, we examine the matrixing procedure for multichannel audio signals. While matrixing is a technology mostly associated with analog sound and considered outdated, it is examined here because it offers a better perspective towards understanding the new technologies in multichannel audio low bitrate coding such as MPEG Surround. At the same time, matrixing is also related to stereo upmixing, which is a technology that is of central interest today. This is due to the fact that upmixing allows all the available 2-channel and monophonic recordings to exploit the potential of multichannel rendering systems. In Section 6, we talk about MPEG Surround, which is a new technology that allows for low bitrate coding of multichannel audio and can be considered as a perceptually motivated matrixing algorithm. MPEG Surround is expected to dominate the field in the future since it achieves bitrates for multichannel audio coding that are comparable to those of MP3 2-channel stereo audio coding. In Section 7 and Section 8, we describe the source/filter and the sinusoidal models respectively. We show how these models, which so far resulted in audio quality degradation at low bitrates, can be applied in multichannel audio recordings and achieve high quality and low bitrate coding for immersive audio applications.

2 MPEG-1 Layer III [2–6]

In this section, we give a basic introduction to MPEG coding of monophonic or 2-channel stereo [1] audio signals. Since our focus is on multichannel audio and given that many tutorials are now available describing MPEG audio coding, we only give a brief introduction here.

The Moving Pictures Experts Group (MPEG) was formed in 1988 by the ISO/IEC organization in order to propose the first international standard in coding of moving pictures and associated audio content. Four years later, in 1992, MPEG-1 coding became an international standard formally known as ISO/IEC IS 11172. The activities of the MPEG-Audio subgroup led to the audio coding part of MPEG-1, ISO/IEC IS 11172-3. The audio coding part of MPEG today finds applications not only in video-related content but also in audio-only applications (the most popular today being portable audio and MP3 encoding, which stands for MPEG-1 Layer III). Next, we describe the basics of MP3 encoding due to its popularity and since the other two layers of MPEG-1 audio (Layers I and II) are similar in philosophy.

The main idea behind MPEG audio coding is to take advantage of principles of psychoacoustics in order to shape the quantization noise so that it will be

[1] In this chapter, the term stereophonic sound refers to 2-channel stereo.

inaudible. Although we are still far from the point of understanding exactly how the human auditory system functions, our knowledge of the human ear physiology and significant efforts of experimental research in psychoacoustics give us an indication of several concepts that have been exploited in MPEG audio coding. More specifically, the idea in MPEG audio coding is that often some sounds can mask others in human perception. Masking can occur when two sounds that are close in frequency occur simultaneously; then, the sound with the higher energy (masker) will mask the lower-energy sound (maskee). Non-simultaneous masking can also occur, when a sound can mask sounds of lower intensity which follow – or even precede – in time. Masking depends on whether both sounds are tonal or noisy relative to each other. The case that a sound might be so low in energy and thus inaudible (depending on a frequency-dependent hearing threshold) is also considered during the coding procedure.

2.1 Principles of Psychoacoustics

Absolute Threshold of Hearing. The intensity of an acoustical stimulus is usually measured in dB SPL (Sound Pressure Level), which is the relative level of sound pressure of the stimulus using an internationally defined reference. The absolute threshold of hearing is expressed in dB SPL and gives us the minimal value of intensity for a tone so that it can be heard in a noiseless environment. This threshold has been found to depend on the frequency of the tone, and has been experimentally found to be approximated by the following relation

$$T(f) = 3.64f^{-0.8} - 6.5e^{-0.6(f-3.3)^2} + 0.001f^4, \tag{1}$$

where f is the tone frequency in kHz and T is the threshold in quiet in dB SPL. The threshold in quiet with respect to frequency is depicted in Fig. 2. For each frequency bin, if the intensity is less than the absolute threshold, the particular frequency component will not be audible and thus can be neglected, resulting in less information to be coded and consequently in coding gain. Additionally, it can be seen from Fig. 2 that in low or high frequencies, the human perception will be more amenable to quantization noise (as long as it is below the absolute threshold) than in middle frequencies.

Critical Bands. Based on our knowledge of the physiology of the human auditory system as well as from psychoacoustics, there have been assumptions that the human ear processes sounds on a frequency-dependent procedure. More specifically, there is evidence to support the assumption that the human ear acts as a bandpass filter, and each band is processed independently of the others in some respects. These frequency bands have been identified by experiments and are known as the critical bands. The center frequencies of the critical bands are not uniformly distributed in the frequency, and the critical bandwidths depend on the center frequency of the corresponding critical band. The distance between two critical bands is measured in Barks, the distance between two adjacent bands being 1 Bark.

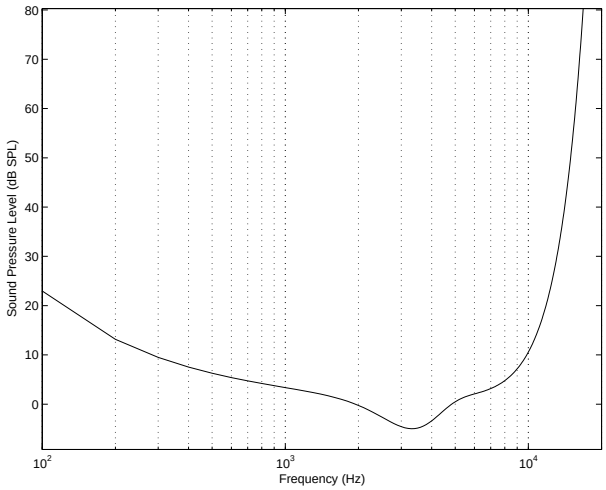

Fig. 2. Absolute hearing threshold in quiet.

Simultaneous Masking. Simultaneous masking is the main masking effect that is considered in audio coding systems. Simultaneous masking corresponds to two or more sounds reaching the eardrums at the same time. For example, a pure tone at a particular frequency will mask all other tonal signals that occur at the same time at the same frequency or within a neighborhood of that frequency. Extensive experiments in this area have revealed specific thresholds for masking depending on the frequency of the masker; if the intensity of a sound is below the corresponding threshold, it will be masked by the masker and will not be audible. In order to derive the masking thresholds, each sound in a recording must be identified as tonal or noise, because different masking thresholds have been derived depending on whether a tone is masking a noisy signal or the opposite (asymmetry of masking). Another important property of the auditory system is the spread of masking, which states that a tone or narrowband (1 Bark bandwidth) noise can mask sounds in adjacent frequency bands. This effect has also been studied extensively and applied in coding applications.

Non-simultaneous Masking. Non-simultaneous masking is also possible in the auditory system, and occurs when a sound of short duration masks lower-level sounds that follow in time (postmasking) and sometimes even preceding sounds (premasking). Postmasking usually occurs for sounds within 50-300 ms of each other, while premasking occurs only for sounds that precede the masker around 1-2 ms. The amount of temporal masking is also frequency-dependent, in addition to its dependency on temporal effects such as the intensity and duration of the masker, and the time interval between the two sounds. Temporal masking has also been studied extensively and its effects can be quantified and used in audio coding.

2.2 MPEG-1 Layer III Coding (MP3 Audio Coding)

MPEG audio coders accept as input PCM coded audio signals with sampling rates of 32, 44.1, or 48 kHz, and the MPEG coded signals are 32-192 kb/s for monophonic and 64-384 kb/s for stereophonic signals, achieving transparency at as low as 128 kb/s for stereo audio (for MP3 audio coding). MPEG audio coding consists of 3 layers (Layers I, II, and III) with increasing complexity and achieved performance in terms of compression.

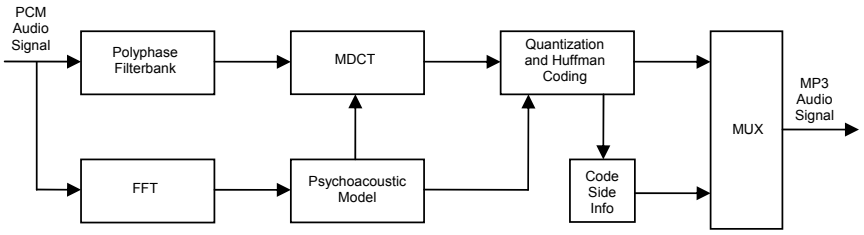

Fig. 3. Block diagram of the MPEG Layer III encoder.

Perceptual Model. The perceptual model for MPEG audio coding is obtained in parallel with the actual coding procedure. In other words, for the same samples that are analyzed at a particular instant at the encoder using the hybrid filterbank that is explained next, an FFT spectrum analysis is applied in order to derive the masking thresholds for each critical band. For MPEG audio, two psychoacoustic models are proposed, Model 1 and Model 2 and they can be used at each of the 3 layers (Model 2 is more complex but also more accurate). In Model 1, a 512 (Layer I) or 1024 (Layers II and III) -point FFT is applied to each block of samples. Then, the frequency components are separated into tonal and noise-like, since the masking asymmetry of the auditory system must be considered. Tonal components are first found, and the remaining (noise-like) components are substituted by a sum of the spectral values per critical band (at a particular frequency which is found as the geometric mean of the noise-like components of each critical band). A spreading function is then applied since masking might influence sounds in adjacent critical bands. Finally, a masking threshold for each subband signal is estimated, taking into account the thresholds in quiet, as explained previously. One problem is that the subband signals obtained by applying the encoding filterbank might correspond to bandwidths different from a critical band. This is especially true for Layers I and II where each subband signal corresponds to $\pi/32$ bandwidth of the hybrid filterbank, *e.g.,* 750 Hz for a 48 kHz sampling rate. In this example, a subband signal at low frequencies will span a multitude of critical bands, while at high frequencies a critical band will be wider than the subband bandwidth. In Model 1, the masking threshold for the subband is found as the minimum of the masking thresholds

of the frequency components that are associated with the frequency range of the subband. This procedure is accurate only for lower frequencies (subband wider than a critical band) but is inaccurate for higher frequencies (subband bandwidth smaller than a critical band). The problem in the latter case is that the noise-like components are concentrated at only one frequency index per critical band, and for subbands within the critical band that are far from the noise component, the estimated masking threshold will be inaccurate. The main difference between Model 1 and Model 2 is that Model 2 does not discriminate sounds in tonal and noise-like and thus is more accurate. In Model 2, a tonality index is only associated with each frequency component, and this index is used to interpolate between tone-masking-noise and noise-masking-tone values. This index is calculated based on a predictability criterion (using prediction from values from two previous windows) since tonal sounds are more predictable than noise-like sounds. Also, in Model 2, the masking threshold for a subband is found as the minimum of the associated critical bands only for the frequencies when the critical band is not much wider than the subband bandwidth. Else, the model calculates the final masking threshold as the average of the thresholds for the frequencies covered by the subband.

Filterbank. The audio samples are processed by a polyphase filterbank in segments of 512 samples. The filterbank is a critically sampled filterbank which divides the spectrum in 32 equal bandwidth subbands, *i.e.,* each subband has bandwidth $\pi/32$ and each filter is centered at odd multiples of $\pi/64$. The filterbank is implemented by modulating a low pass filter $h(n)$ at the specified frequency positions, *i.e.,*

$$s_i(n) = \sum_{k=0}^{511} x(n-k)H_i(k), \tag{2}$$

where

$$H_i(n) = h(n)\cos\left(\frac{\pi(2i+1)}{64} + \phi(i)\right) \tag{3}$$

is the bandpass filter for subband i ($i = 0, \cdots, 31$), $x(n)$ is the audio (PCM) sample at time n ($n = 0, \cdots, 511$), $s_i(n)$ is the corresponding subband sample, $h(n)$ is the prototype window defined by the standard (length of 512 samples), and $\phi(i)$ are the phase shifts for each filter, also defined by the standard. Clearly, each bandpass filter of the filterbank is a modulated version of the prototype window (which in the frequency domain is a low-pass filter) at the corresponding center frequency. The filterbank is not perfect reconstruction, *i.e.,* it introduces a (small) distortion to the audio signals. This filterbank is used in all 3 layers of MPEG. The filterbank achieves aliasing cancellation with the proper choice of prototype window and phase shifts.

For layer III coding, the polyphase filterbank is followed by a MDCT (Modified Discrete Cosine Transform) filterbank [7], a perfect reconstruction filterbank, which offers improved frequency resolution compared to Layers I and II. This

is important because the polyphase filterbank follows a uniform division of the frequency domain which is in contrast to the critical bandwidths. The MDCT window size is either 36 or 12 samples (with 50% overlapping it corresponds to 18 and 6 subband samples, respectively), the smaller value corresponding to the detection of transient sounds, in order to alleviate the pre-echo problem (window switching). With the long window, the MDCT procedure results in dividing the frequency range in $\pi/32/18$ or $\pi/576$ (42 Hz for 48 kHz sampled audio) equal bandwidth bands, which is a significant improvement in frequency resolution. The consequence of poorer time resolution which might practically result in pre-echo distortion is alleviated by using the short window for transient signals.

Pre-echo. In audio coding, the PCM signal is processed in short-time segments (or frames), usually in the order of few ms, and each segment is coded individually. In audio coding methods that exploit the perceptual masking effect for shaping the quantization noise, transient parts of the audio signal result in audible coding noise which is often referred to as pre-echo. The problem stems from the windowing procedure of the audio signal in short segments and the fact that the quantization noise is shaped using the spectral properties of the entire segment. The problem is clear in the example of a sharp increase in the amplitude of the audio signal which occurs in the middle of a segment. The masking thresholds for each segment are estimated based on the frequency response of the entire segment. This means that the higher energy part will dominate the segment and will result in using higher masking thresholds than those that would be estimated if only the first low-energy part of the segment was present. This consequently results in audible noise (pre-echo), since the masking thresholds derived are not correct for the audio segment. Clearly, this problem can be avoided if very short windows are used for segmenting the audio signal. However, the better the time resolution, the worse the frequency resolution will become, so using very short segments results in less coding gain. An apparent solution is to use short windows only when signal transients are detected, and this is actually performed in MP3 and AAC coding (window switching). Additionally, in AAC Temporal Noise Shaping is also applied. These issues are discussed further in the following sections.

Quantization and Entropy Coding. We describe the coding procedure for Layer III coding. After the MDCT procedure, the subband samples are organized in scale-factor bands, each of which is approximately of critical bandwidth. The use of the perceptual model and the derived masking thresholds are applied at this stage in an analysis-by-synthesis iterative manner, so that the quantization noise will be below the masking thresholds using the smallest possible number of bits per subband. In other words, the encoder must calculate the quantization noise (the difference between the original spectral values from the quantized value) for a given set of parameters and repeat the procedure until the best bitrate is achieved while keeping the noise below the masking thresholds. Non-uniform quantization (power law) is employed and Huffman coding is applied to

the quantized values. The procedure is applied per subband (scale-factor band) and for each block of samples.

More specifically, the subband samples are organized in groups, each of which corresponds approximately to a critical bandwidth. Each group is assigned a scale-factor and thus termed scale-factor bands. Initially, each scale-factor is equal to 1.0. The coding algorithm is an iterative procedure of two nested iterations, where the inner loop controls the coding bitrate (bit allocation and Huffman coding) and the outer loop controls the noise shaping. Noise shaping is achieved using the scale-factors and analysis-by-synthesis for estimating the quantization noise. If the quantization noise is found to exceed the masking threshold of a particular scale-factor band, the scale-factor is increased, so that the values of the subband samples are increased which translates into using more bits for the particular scale-factor band. One issue with MP3 coding is that the iterative coding procedure might not converge, so special considerations to avoid this problem must be made in practice.

3 MPEG-2 AAC [8–11, 5, 6]

The second phase of the MPEG Audio coding standardization was finalized in 1997. Initially, in 1994 MPEG-2 BC (Backwards Compatible) and MPEG-2 LSF (Low Sampling Frequencies) was standardized as ISO/IEC IS 13818-3. MPEG-2 BC was proposed as a multichannel audio extension of MPEG-1 Audio, providing high-quality audio (CD-like quality) at rates 640-896 kb/s for five audio channels. MPEG-2 LSF was introduced for sampling frequencies that are lower than what MPEG-1 Audio allowed for (16 kHz, 22.5 kHz, 24 kHz). However, in 1994 it was proposed that a newer standard should be introduced, which would be more efficient in term of datarates for high quality multichannel audio, by relaxing the constraint of backwards compatibility with MPEG-1 audio. The result was the MPEG-2 NBC (Non-Backwards Compatible) algorithm, known as MPEG-2 AAC (Advanced Audio Coding), formally known as ISO/IEC IS 13818-7. MPEG-2 AAC allows for coding of five full-bandwidth audio channels in transparent quality at 320 kb/s rate. The supported sampling rates vary between 8-96 kHz. Below we give a brief description of AAC with reference to the MPEG-1 coding algorithm, since AAC is actually an improvement of MPEG-1 audio coding and the two methods have a lot of similarities.

AAC operates in three different profiles, offering a tradeoff between audio quality and complexity requirements. These three profiles are the Main Profile, the Low Complexity (LC) profile, and the Sample Rate Scalable (SRS) profile, the main profile being the highest quality profile. Here, we focus on the main profile. We mention that the main configurations in AAC are monophonic, 2-channel stereo, and 5.1 channel (5 channels and LFE (Low Frequency Effects) channel). However, the standard supports up to 48 audio channels. The description of the components of AAC follows.

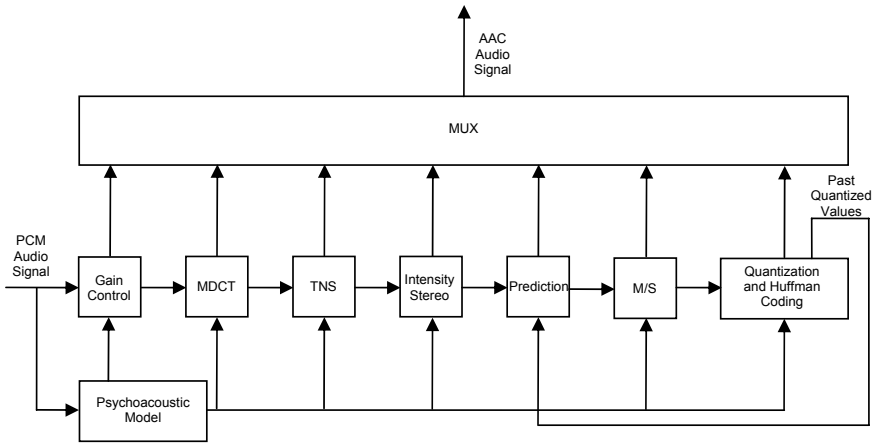

Fig. 4. Block diagram of the MPEG AAC encoder.

Perceptual Model. The Model 2 perceptual model of MPEG-1 Audio was adopted for AAC.

Filterbank. The filterbank used for AAC is an MDCT filterbank with very high frequency resolution. The window used is a 2048 sine or KBD window (explained next), which translates into using 1024 samples of audio for each segment (21 ms time resolution and 23 Hz frequency resolution for 48 kHz sampling rate). The low time resolution of the window might result in pre-echo problems (see MPEG-1 Audio section), so for transient signals the window length is switched to 256 samples (2.6 ms time resolution and 187 Hz frequency resolution at 48 kHz sampling rate). The sine and KBD (Kaiser Bessel Derived) windows are used depending on the signal at each time frame. When the signal contains strong frequency components with separation below 140 Hz, then the sine window is preferred, while when the strong frequency components in the signal are separated by more than 220 Hz, then the KBD window is used. The window shape can change from frame to frame, although the first half of the current window must follow the shape of the previous window.

Window switching. As in MP3 audio coding, when transients are detected in the audio signal, the MDCT window length switches from 2048 to 256 samples for avoiding the pre-echo problem. It is possible that in one channel the short window is used while at another channel the long window is used, at the same time instant. Then, in order to maintain synchronization between the multiple audio channels, the audio samples that are windowed by the short window are organized in blocks of 8, so that these will remain of the same sample size as in the case of the long window.

Prediction. For each of the spectral coefficients that are the output of the MDCT filterbank, a predictor is used for improving the coding gain. It is known that encoding the prediction error instead of the actual values of a signal improves the coding gain for stationary signals, since the prediction error is a "whitened" (*i.e.,* decorrelated) version of the original signal. The predictor is applied in AAC, only for samples of the audio channel that have been processed by the long window of the MDCT, since there would be no benefit in coding when using the predictor for transient signals. Prediction is based on previous values of the corresponding spectral coefficient, *i.e.,* for estimating the prediction coefficients for each spectral component, the corresponding components of previous frames are needed.

The predictor used is a 2^{nd} order predictor, which is adapted to the statistics of each signal frame by employing an LMS (Least-Mean Squares) -type adaptation procedure. The predictor is applied to spectral components that correspond to frequencies up to 16 kHz. The predictor is applied only when there is a coding gain that justifies the overhead of prediction, which is determined for each scale-factor band (predictor control). For the spectral coefficients that the predictor is applied, the prediction error is coded instead of the actual spectral values and is transmitted to the decoder along with the prediction coefficients.

Quantization and Coding. The coding procedure in AAC is similar to MP3 coding. Again, non-uniform quantization and Huffman coding are applied in an iterative manner, with two nested loops. Scale-factors are again applied as in MP3 coding, and are modified in steps of 1.5 dB at each iteration. In total, 49 scale-factor bands are used, and scale-factors for each band are PCM coded.

TNS. The idea of TNS (Temporal Noise Shaping) is based on the duality between the time and frequency domains, and can be thought of as predictive coding applied to the spectral domain. TNS is applied in order to better address the pre-echo problem, which is not sufficiently treated with window switching. For example, strongly pitched signals (such as voiced segments of speech signals) are not efficiently coded using the short MDCT window. This happens because such signals have the form of an impulse train, which might need a long sequence of short windows for avoiding the pre-echo problem. Use of the short window, though, is not efficient in terms of coding gain and at the same time results in overhead.

For transient signals, the problem with pre-echo is related with the fact that within a window the signal envelope changes considerably. Thus, it would be desirable to "flatten" the time domain envelope of the signal within a frame. It is known that in the frequency domain, use of prediction achieves such a whitening operation on the signal, by encoding its prediction error which is much "flatter" in the frequency domain compared to the spectrum of the original signal (signal "whitening"). Based on the duality between the time and frequency domains, TNS is performed by applying prediction to the frequency domain of the signal frame, which means that the prediction coefficients will capture the

time-domain envelope of the signal, and that the prediction error will correspond to a "whitened" version of the time-domain signal. Thus, coding is applied to this prediction error and along with the prediction coefficients the signal can be reconstructed at the decoder. The use of the MDCT filterbank along with the prediction module (whitening in the spectral domain) and the TNS module ("whitening" in the time domain) are the main novelties which offer significant improvement in terms of coding gain in AAC compared to MP3 coding.

Joint Stereo. Joint stereo coding in AAC (but also in MP3 audio coding) includes both M/S coding and Intensity Stereo Coding, which are described in the following section. These are methods that exploit interchannel similarities for improved coding gain. The description that follows explains how these algorithms operate on a pair of channels. In the case of multichannel audio, channels are arranged in pairs symmetrically on the left/right axis (*i.e.,* left and left surround, right and right surround, center and LFE). However, in Intensity Stereo Coding, channel coupling is also proposed (generalized ISC), which in essence allows for intensity stereo coding between pairs that are not necessarily symmetrically placed with respect to the left/right axis (depending on a rate optimization and distortion minimization procedure). Also, channel coupling allows for downmixing additional sound objects to the stereo image, *e.g.,* adding a voice-over to an existing multichannel recording.

Extensions of AAC AAC is rapidly being replaced in commercial applications (such as DTV) by AAC+. AAC+, formally known as (High Efficiency) HE-AAC is a combination of AAC coding with Parametric Stereo (described in Section 6.2) and Spectral Band Replication (SBR). SBR belongs to a family of methods known as bandwidth extension methods, where the objective is to "predict" in some sense the high-frequency component of an audio signal given its low-frequency counterpart and possibly some side information. Description of SBR is beyond the scope of this chapter. AAC+ significantly improves the coding gain of AAC, achieving bitrates in the order of 160 kb/s for high-quality 5.1 multichannel audio.

4 Joint Stereo Coding

4.1 M/S Coding of Stereo Audio [12]

The principles of perceptual audio coding that were described previously, have been based on the assumption of a monophonic audio signal. In the case of stereophonic reproduction, two different audio signals are reproduced through the loudspeakers or headphones. In this case, there are additional constraints regarding the masking thresholds because in binaural hearing (*i.e.,* when different audio signals are presented to the listener's ears) some psychoacoustic principles must be considered. For audio coding, the focus is on the fact that in binaural hearing the masking thresholds have been found to decrease when the

masking signal and the quantization noise are out of phase [13]. In other words, in a stereophonic (or multichannel) setting, human hearing improves due to the phase differences between the sounds at the two eardrums, and the quantization noise at some frequencies might be unmasked compared to the monaural case (binaural unmasking). In turn, this implies that the masking thresholds are lower in this case, and have been found to decrease as low as 15 dB compared to the monaural masking thresholds for broadband noise (or even higher for narrow-band noise). The effect is known as Binaural Masking Level Difference (BMLD) and is concentrated to lower frequencies (up to 2 kHz). In higher frequencies, auditory perception is based on the signal envelopes (this fact is exploited for audio coding by Intensity Stereo coding as explained later in the section).

For audio coding applications, BMLD implies that the masking thresholds used for perceptual coding of monaural signals must be modified in order to accommodate stereo signals. In a different case, *i.e.,* if a 2-channel stereo signal is encoded as two independent monophonic signals without considering BMLD, quantization noise might be audible. In fact, it is described in [12] that in listening tests, a 64 kb/s monophonic signal was rated better than a stereo 64 kb/s/channel stereo signal in terms of quality. In other words, the effect of BMLD is that a stereo recording requires a higher bitrate for high-quality encoding than twice the bitrate of a monophonic recording of the same quality. In order to address this important problem, sum-difference coding of stereo audio was proposed in [12]. The method is also referred to as M/S coding (mid/side coding), and has been implemented as part of MPEG-1 Audio, MPEG-2 AAC, and Dolby AC-3 coders, among others.

The basic idea behind M/S stereo coding is to reduce the increased bitrates for stereo audio coding using a transformation that takes advantage of the redundancy between the two stereo channels. The idea is to replace (in each audio subband) the left (L) and right (R) signals by a sum (middle, M) and difference (side, S) signal that are related with the original left and right channels with the following relations

$$M = \frac{L+R}{2}, \quad S = \frac{L-R}{2}. \tag{4}$$

The above transformation is applied only for low frequency sounds (below 2 kHZ), and only when the masking thresholds found by the perceptual model for the left and right channel are within 2 dB. This means that there is need for a switching mechanism between the dual mono coding of the Left and Right channels, and the M/S coding mode.

A first note is that the above transformation is a KLT (Karhunen Loeve Transform) -type [14], with eigenvectors $(1/\sqrt{(}2))[1\ 1]^T$ and $(1/\sqrt{(}2))[1\ -1]^T$. Thus, this would be an exact KLT transform if in a scatter plot of the L and R samples, the main axis was rotated 45 degrees counter-clockwise, or in other words, the spatial image (phantom source) of the stereo sound was exactly between the two channels. The reader is referred to the example of Fig. 6, where a scatter plot of the samples from the Left and Right channels is plotted for a short segment of a stereo music recording, along with the directions of the two

eigenvectors (dashed lines). The angle of the primary eigenvector with respect to the Left channel is approximately 45 degrees. This is usually the case in practice, so in fact M/S coding can be considered an "approximate" KLT transformation. In this sense, it is of interest to note that this is a fully invertible transform, *i.e.* the L and R signals can be easily and fully obtained from the M and S signals. A simpler view of the M/S transformation is to consider the extreme case when L and R would be equal (maximal correlation); then S would be zero, thus we would need to encode only M, which would result in 50% gain in bitrate. In practice, this limit cannot be accomplished, but the more correlated L and R are, the more the coding gain from M/S coding. In addition to the coding gain for M/S coding, there is a re-calculation of the masking thresholds for this transformation, in order to account for the BMLD effect. Finally, we mention that this procedure can also be considered as a matrixing procedure (information about matrixing and the KLT can be found in Section 5).

4.2 Intensity Stereo Coding [15]

Intensity stereo coding can be considered as a counterpart for M/S coding for high frequencies (above 2 kHz). It has been implemented as part of MPEG-1 Audio, MPEG-2 AAC, and Dolby AC-3, among others. It is based on the fact (related with the Duplex theory discussed later in Section 6.1) which states that for high frequencies, the level difference between the sounds at the two eardrums (Interaural Level Difference - ILD) is the most important cue for sound localization. Thus, in order to recreate a stereo (spatial) image, it would suffice for the two channels to contain the same audio signal, and scale each channel with appropriate scale-factors. Intensity stereo coding operates based on this philosophy, by extracting the energy for each subband (critical band) for each audio channel. By considering this procedure as it evolves in time, the method retains only the time-domain envelope of each subband signal, instead of the actual subband samples. If viewed for a particular time frame, the method actually retains the spectral envelope only of each channel (the energy at each critical band). The following procedure is applied for each subband signal (scale-factor band equivalent to critical band), considering the stereo case. The left and right signals are summed (as in the M signal of M/S stereo coding) producing a downmixed signal (we remind the reader that this occurs for frequencies above 2 kHz). For each subband of the left and right channels, the energy is also calculated for each time frame (one value per subband). In the decoder, the donwmixed signal is analyzed in the subbands, and each subband is multiplied with the corresponding factor for the left and right channel. In this way, the energy envelope for the two channels is retained, and the stereo image is successfully reproduced. We should note that this procedure is valid for recreating the spatial image and not the waveform itself, and this is why the downmixed waveform is preserved.

As in M/S coding, intensity stereo coding can also be considered as a KLT-type transformation. Considering again the fact that for most stereo signals the phantom image will be in the middle between the two channels for most of the time, this corresponds to an angle of 45 degrees in the scatter plot, and to the

$(1/\sqrt{(2)})[1\ 1]^T$ and $(1/\sqrt{(2)})[1\ -1]^T$ eigenvectors. The KLT signals would be the sum and difference signals. If we wanted to reduce the amount of information using KLT (as in Principal Component Analysis - PCA), we would retain only the sum signal (corresponding to the higher eigenvalue, *i.e.,* the higher energy), while we would not consider the difference signal. This is exactly what happens in intensity stereo coding (it would be impractical to calculate and code the exact angle for the eigenvectors for each time frame). The fact that we can retain the stereo image while considering only one of the two KLT components is possible due to the way the human auditory system operates (and to the fact that we also retained the energy values). This procedure of downmixing and side information extraction of intensity stereo coding is in fact the basis for spatial audio coding that we examine later in the chapter. Similarly to M/S coding, Intensity Stereo coding can also be considered as a matrixing procedure.

5 Matrixing

Matrixing in general as a term describes the procedure when a multichannel sound recording is downmixed in a new recording containing a smaller number of channels. The origin of matrixing has been the technology in the film industry. More specifically, multichannel surround sound systems have been a result of advancements in the sound of the movie theaters, when it was clear early on that surround sound is necessary for experiencing realism when watching a movie. The 1940 movie "Fantasia" is the first example of a movie containing surround sound. The technology was based on 4 audio channels, and was known as "Fantasound". However, new film formats that emerged (35 mm film) did not have enough capacity for multiple channels of sound and, in fact, for a period during the 1970s film sound was monophonic. The fact that there was actually space in the movie formats for 2 channels of audio gave birth to matrixing. The idea was to downmix multiple sound channels into the two channels that were available. In this technology, Dolby assumed a leading role, developing Dolby Stereo and Dolby Surround, followed by Dolby Pro Logic and currently Pro Logic II. It is interesting to note that nowadays matrixing is mainly used for a different application than its origins, namely upmixing. Upmixing refers to applications when a 2-channel stereo recording is processed so that it can be reproduced through a multichannel system. Dolby Pro Logic II, Harman Kardon Logic 7, DTS Neo are technologies among others that perform this task. Upmixing is based on the matrix decoders and will also be examined later in this section. In this section, we describe the Dolby encoder as an example of matrixing due to its popularity. We also describe KLT-based methods for upmixing and multichannel audio coding. KLT is not strictly a matrix-type transformation since the number of channels after the transform is not reduced. However, its decorrelating property can be used for upmixing, by separating sounds in phantom and ambient sources, and for matrix-type loss concealment when eigenchannels are ordered in decreasing importance similarly to the manner that low-rank modeling operates.

5.1 Dolby Stereo and Dolby Surround [16]

Dolby Stereo contains 4 channels of audio, Left, Center, Right, and Surround. These are then matrixed into 2 channels using the following procedure, known as Dolby MP (Motion Picture) encoding or sometimes referred to as Dolby Stereo (matrix) encoding. The Center channel is reduced by 3 dB and then added both to the Left and Right channels. The 3 dB reduction is needed for maintaining constant total power. The Surround channel is also reduced by 3 dB, it is filtered with a bandpass filter retaining only the frequencies between 100 Hz to 7 kHz, and then added to the Left and Right channels but with a +- 90 degree phase shift. Specifically, it is added with a +90 degree shift to the Left channel and with a -90 degree shift to the Right channel. The above procedure is shown in Fig. 5. The two channels that are the output of the encoder are referred to as the Left Total and Right Total channels, in order to distinguish them from the original (before matrixing) Left and Right channels.

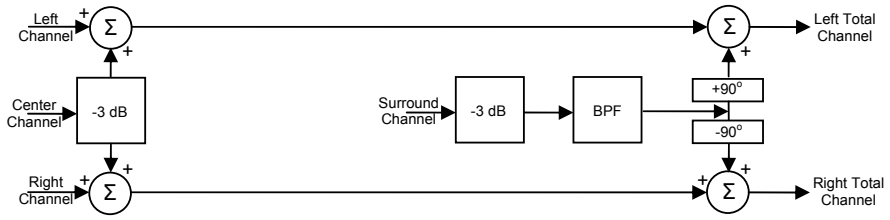

Fig. 5. Dolby Stereo matrix encoding.

One of the important constraints of matrix encoders is backwards compatibility. The Dolby Stereo encoder is designed with this constraint in mind, so that when the matrixed Left Total and Right Total channels (which in reality contain 4 multiplexed channels) are reproduced through a 2-loudspeaker system, they will sound as 2-channel stereo. This can be easily seen when considering a listener that is seated symmetrically with respect to the 2 loudspeakers (the stereo "sweet-spot"). For a listener seated at the sweet spot, it is known that a "phantom image" is created between the two loudspeakers, when the 2 audio channels contain the exact same audio signals, because the audio signals arrive at the listener's ears with equal amplitude and phase difference (zero interaural level and phase difference). Regarding the Center channel, since the two matrixed channels contain the Center channel in the same manner (equal power and no phase difference), it will indeed be heard as a phantom image from the center between the two loudspeakers. It is of interest to note that in films the Center channel contains mostly dialogue, which is best reproduced using a loudspeaker in the center of the screen. Common parts of the original Left and Right channels will also be heard from the phantom image, exactly as it happens in 2-channel stereo. On the other hand, the 180 degree difference of the Surround

channel between the Left and Right output signals completely avoids any leakage of the Surround channel to the phantom image. At the same time, this phase difference between the Left and Right outputs regarding the Surround channel will create a diffuse nature to the sound, which is exactly what is expected by the surround sound. Thus, matrixing actually enhances stereo playback even when only 2 loudspeakers are present.

The main functionality, though, of matrixed sound is that it can be decoded and reproduced by a multi-channel playback system. For home-theater applications, which are mostly related to intelligent environments (as opposed to cinema sound), Dolby Surround was the initial decoder that Dolby produced, which is a passive decoder. The Center channel is extracted as the sum of the Left and Right matrixed channels, and the result is the same as in the phantom image case that was described above. So, for the front channels Dolby Surround decoding does not involve any processing, and these channels are reproduced exactly as in 2-channel stereo. It must be noted that in some Dolby Surround systems, it is possible to add a loudspeaker at the front center position, and not rely on the "phantom" stereo image. The addition of a Center loudspeaker avoids the problem that the "phantom" image is valid only when the listener is seated at the sweet-spot position, so that the farther one is seated from this position, the less evident the center image becomes. The disadvantage, however, of using a Center loudspeaker when creating a Center channel from Dolby encoded material is that this channel is created by adding the left and right channels, so channel leakage will occur. While this summation will result in cancellation of the Surround channel from the Center channel (since it has been added to the Left Total with a +90 degree phase difference and to the Right Total with -90 degree phase difference), the Center will still contain the sum of the Left and Right channels. This is avoided when relying on the "phantom" image. At the same time, the Center channel remains in the Left and Right channels. Thus, the addition of the Center loudspeaker will provide a better localization of the Center channel for off-center listener positions, but overall will result in a "narrowed" spatial image, due to channel leakage between the Left, Right, and Center channels.

The Surround channel is extracted by subtracting the Right Total channel from the Left Total, and also delayed to exploit the precedence effect (explained below). This procedure results in a Surround channel with a completely canceled Center channel as desired, containing the initial Surround channel (before matrixing) and the difference between the Left and Right channels. This difference is acceptable for a Surround channel to contain, since the more correlated the Left and Right channels are, the more uncorrelated the difference is with the sum signal that comes from the phantom image, which offers a diffuse effect to sound. Thus, there is (almost) perfect separation between the Surround and the Center channels after the decoding process. Additionally, the bandpass filtering of the Surround channel during encoding and decoding improves the separation from the other channels and is acceptable since the Surround channel is viewed as an "effects" channel, in the sense that it is only supplementary compared to the front channels. It is also important to explain the use of the delay during

decoding. This delay (in the order of 10 ms), is added to the Surround channel to exploit the precedence effect. The precedence effect states that when humans listen to an identical sound from two different sources, they perceive the sound as coming from the direction of the sound that first reaches the eardrums. Based on this effect, even if there is non-perfect separation between the channels, the front sources are given more emphasis, so leakage effects are minimized.

5.2 Dolby Pro Logic [16]

One of the main problems of the Dolby Surround decoder is the fact that it relies on the "phantom" stereo image to create the Center channel. Also, the Left and Right channels contain the Surround channel by only a 3 dB separation (while the 180 degree phase difference provides a perfect in principle separation of the Surround with the Center channel). At the same time, the decoded Surround channel contains the difference of the Left and Right channels, which in general should be considered as channel leakage, even if in some cases it is not an audible problem. Finally, the separation between the Surround and the Center channels is only theoretical, and in practice it is possible that the level of the Left Total and Right Total channels might not be exactly equal, which would create a leakage effect (crosstalk) of the Center to the Surround and vice versa. Some of these problems are ameliorated with the use of the bandpass filter and the time delay in the Surround channel, but still the Dolby Surround encoder is far surpassed in quality (regarding the spatial image that is created) by the Dolby Pro Logic decoder.

The main concept in the Dolby Pro Logic decoder is directional enhancement. The decoder at any time instant identifies a sound as being the prominent in the recording (if such a sound exists). Based on amplitude differences between the Left and Right Total channels, and their sum and difference channels, the decoder is able to derive a direction of dominance, in a manner explained next. This is achieved by enhancing the Dolby Surround decoder, which is a passive decoder, with an adaptive matrix based on Voltage Controlled Amplifiers (VCAs), which have the objective of enhancing the soundstage towards the direction of dominance. The use of the adaptive matrix is the reason that the Pro Logic encoder is referred to as an active decoder.

In order to detect the direction of the image at a particular time instant, the decoder is based on finding the logarithmic difference between the absolute voltage of two pairs of signals. The one pair is the Left Total and Right Total channels. These channels, in addition to the Left or Right channels, they contain both the Center and Surround channels with equal power. Thus, the difference between the Left Total and Right Total channels gives an indication regarding the Left/Right dominance of the sound at a particular instant. The second pair of sounds that are compared is the sum signal (Left Total + Right Total) and the difference signal (Left Total - Right Total). The sum signal contains the sum of Left and Right in addition to the Center, while the difference signal contains the difference between the Left and Right in addition to the Surround channel. The difference between the sum and difference signals gives an indication of the

dominance regarding the front/back (Center/Surround) direction. The result from these operations is used as the control input to 8 VCAs which are used to control the enhancement of the passive decoder (2 signals to be controlled – Left Total and Right Total, 2 control signals for Left/Right dominance, 2 control signals for Center/Surround dominance).

The adaptive matrix succeeds in its operation by exploiting the following concepts:

- Cancellation concept. Considering for example that the dominant direction is the center direction, enhancement to the passive decoder can be achieved by canceling the Center channel that is inherent in the Left and Right channels. For this purpose, a simple solution is to add the opposite of the Right channel to the Left and vice versa. This procedure will cancel the Center channel in the left and right loudspeakers, it will however add the opposite of the Right to the Left which is in fact a leakage effect.
- Constant power concept. In the case that there is a change in the prominent direction in the sound recording, the decoder will abruptly change the enhancement towards the new direction. By keeping a constant power for the recording, the decoder achieves the changes in direction without abrupt changes in the overall intensity of the recording, which would create an unstable image to the listener. It is important to mention that often no dominant signal exists in a recording, and also there are several degrees of dominance (which are related with the amount of dominance in the power difference at an instant between the various channels). So the decoder acts appropriately, enhancing a direction only to the degree required by the recording, or sometimes operating at a fully passive mode.

5.3 Dolby Pro Logic II [17]

While matrix encoding is becoming a technology of the past, Dolby introduced the Dolby Pro Logic II decoder, which is an essential component of all current home-theater systems. The Pro Logic II decoder is very similar in structure with the Pro Logic encoder, with the only difference that it provides sound to be reproduced based on the 5.1 specifications, in other words it derives a stereo Surround channel and also a LFE channel. Another difference compared to the Pro Logic decoder is that the Pro Logic II decoder does not filter the Surround channel with a bandpass filter, allowing for full-bandwidth Surround effects. The reason that Pro Logic II has become very popular even now that matrix systems have been replaced by discrete multichannel coding systems, is that Pro Logic II decoders can produce a convincing surround sound effect from conventional 2-channel stereo or even monophonic recordings. In other words, the consumer can utilize his multichannel reproduction system even when listening to music recorded on a stereo CD or when viewing films only encoded in conventional stereo format. While Pro Logic II is very similar in structure to Pro Logic, it has actually been developed with this stereo upmixing concept in mind, which proved to be very popular for home-theater consumers.

5.4 KLT-based Matrixing

Multichannel audio signals (including 2-channel stereo) can be viewed as realizations of a multivariate random process. Stereo audio for example can be viewed at each time instant as a 2-coefficient vector (the samples from the Left and Right channels). Since KLT is an optimal vector decorrelation method, its application to coding of stereo sound has been considered early on. As we saw, M/S coding originates from the KLT transform, considering that the eigenvectors remain constant, with the primary eigenvector at a 45 degree angle with respect to the initial Left/Right coordinates. This simplification is based on the concept that the main direction for the stereo audio is the "phantom" image at the center between the two loudspeakers, which is approximately the case in practice. For better understanding why the 45 degree angle corresponds to the stereo "phantom" image, an explanation is provided later in this section (when KLT upmixing is described). The reason for not applying the actual KLT decorrelation is that KLT is a signal-dependent transform, which means that at each time instant the angle of the eigenvectors would change. In this case, in order to be able to reconstruct the audio signals at the decoder, the angle of the primary eigenvector at each time instant would need to be transmitted by the encoder, which would cancel the coding gain we might have by the decorrelation process. Consequently, in practice it is not trivial how to exploit the advantages of KLT, and M/S coding proved to be an effective method for approximating the KLT with constant eigenvectors for 2-channel stereo.

KLT for multichannel audio [18]. An attempt to apply KLT decorrelation as an alternative to M/S coding in AAC has been described in [18]. The Modified AAC KLT (MAACKLT) method achieves multichannel audio decorrelation via the KLT by re-estimating the eigenvectors every few frames of the audio signal and not at each time instant. In this case, the decorrelation is not perfect since the eigenvectors are not exactly correct for each time frame, but the method has been shown to achieve good performance by re-estimating the eigenvectors every few hundreds of frames.

Strictly speaking, KLT is not a matrixing method because the number of eigenchannels (KLT transformed channels) is equal to the number of the original audio channels. For a 5-channel recording, there are 5 eigenchannels. The coding gain is due to the decorrelation of the 5-dimensional vector containing the audio samples at each time instant. It is of interest that the KLT is applied in the frequency rather than the time domain, for avoiding time-delay and reverberation that is inherent in the time domain. In a broader sense, the MAACKLT method is a matrixing method since it arranges the various channels in order of importance, and in case of traffic delays and packet losses, a subset of the eigenchannels can be used for reconstructing the original number of channels at the decoder. In this case the algorithm becomes a matrix-type method, since *e.g.,* 5 channels are encoded into less than five (for a packet loss) and these are decoded again at the receiver into 5 channels again (with possible audible distortion). In [18], contrary to what it might be expected, the eigenchannels are not

ordered based on the eigenvalue (energy-based) criterion, but on their relation to the ordering of the initial channels. It is proposed that for a 5.1 configuration, the ordering of channel importance (decreasing) should be (i) Center, (ii) L/R pair, (iii) Ls/Rs pair, (iv) LFE channel. The eigenchannels cannot be associated with a single audio channel, however the authors propose that, based on their experiments, the first eigenchannel is more related with the Center channel and so forth. Thus, the ordering in importance of the eigenchannels is based on their order following the initial channel order and not the related energy. When packets are lost, due to the different significance given at the encoder to each eigenchannel, they will correspond to one of the least important eigenchannels. In the decoder, the multichannel recording is reconstructed using the inverse KLT of the remaining eigenchannels. For the pairs of the same significance, a simple concealment method is proposed.

KLT for Stereo Upmixing [19]. As we described, Dolby Stereo allows encoding of multiple channels into two channels, and decoding again into multiple channels at the decoder (using Dolby Pro Logic). Dolby Pro Logic II – based on the same decoder – can upmix a stereo recording for multichannel rendering. Similarly, KLT can be used for matrixing existing multichannel recordings as explained in the previous paragraph, but it can also be used for stereo upmixing. In this section we describe one method that can achieve such a result.

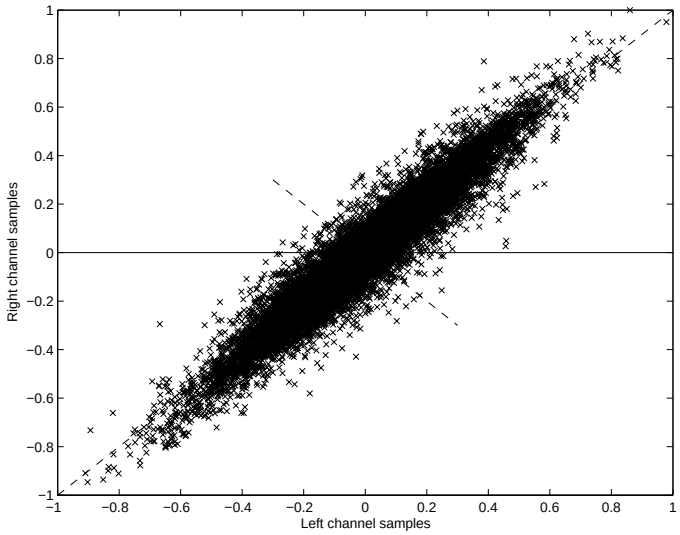

Fig. 6. Scatter plot (left *vs.* right channel) of a short segment of a stereo recording.

The idea is that in a stereo recording it is not difficult or costly in bitrate (as it is in multichhanel audio coding) to follow the angle of the primary eigenvector

for each time instant. An adaptive method for doing this without significant computational cost is described in [19]. For 2-channel stereo, if we are given the angle of the primary eigenvector then we know the exact eigenvectors (since they are perpendicular and unit-length). If the Left and Right stereo samples at time k are $x_L(k)$ and $x_R(k)$, the corresponding eigensignals will be $y(k)$ (corresponding to the main direction – largest eigenvalue) and $q(k)$. The primary eigenvector is denoted as $\boldsymbol{w}(k) = [w_L(k)\ w_R(k)]^T$, and the angle of the primary vector is $a(k) = \arctan(\frac{w_L}{w_R})$. In Fig. 6, an example scatter plot (Left channel samples *vs.* Right channel samples) is given, for a segment of a stereo recording. In the figure, the directions of the two eigenvectors are also indicated using dashed lines. It is interesting to note that the primary eigenvector (angle of dominance) has an angle of around 45 degrees relative to the Left channel, which is usually the case in practice. The fact that the dominant direction is located in the middle of the two channels indicates that this direction corresponds to the "phantom" stereo image, while the secondary direction corresponds to the signal ambience. This in turn implies that the KLT primary component y will correspond to the "phantom" image while the secondary component q will correspond to the ambience. The fact that the two components are uncorrelated further supports these assumptions. In order to better explain why the 45 degree angle corresponds to the "phantom" image, it is useful to consider the extreme case when the Left and Right channels contain the same signals. Then, the stereo "phantom" image will contain the total audio recording, since the Left and Right channels will arrive at the eardrums of a listener located to the "sweet-spot" position with no interaural level and time difference. If we plotted a scatter plot of the Left and Right channels at this extreme case, all points in the two-dimensional space would fall on a single line, which would have 45 degree angle with respect to the x-axis (the line $x = y$).

The idea in [19] is to upmix the two uncorrelated components y and q with a 5×2 matrix which will create the desired 5 channels. In practice, the upmixing matrix is a 4×2 matrix creating a mono surround channel, which is converted in a stereo surround (required for 5.1 systems) by a decorrelation procedure. Thus, the main problem is to obtain the 4×2 upmixing matrix. The design of the matrix is based on the fact that the y component that corresponds to the main direction is the "phantom" image component and thus should be the input to the center loudspeaker, while the q component corresponds to the secondary direction (smaller energy content) which is thus the ambience signal and should be sent to the surround loudspeakers. The signals sent to the left and right signals are found based on a linear combination of y and q. The procedure is similar to inverse KLT, however depending on a Left/Right energy domination, only the dominant channel is found as the linear combination of y and q while the other channel is a weighed version of q only). Additionally, the signals for all channels are weighed by the amount of correlation between the left and right channels so that strongly correlated signals correspond to placing emphasis in the front channels, while weakly correlated signals result in a more diffuse reproduction, exactly as desired.

6 MPEG Surround [20]

MPEG Surround is a recent standard by the MPEG Audio group on Spatial Audio Coding (SAC). This is a new approach that will allow for multichannel audio coding in rates that are similar to - and will be backwards compatible with - mono or stereo MP3 coding. In SAC, a sum signal is created from the various audio channels, and is the only audio signal transmitted along with small side information (in the order of few kb/s). In the receiver, this side information is applied to the downmixed signal in order to produce the necessary cues that will result in correct spatial rendering of the original multichannel recording. SAC originated from the work in Binaural Cue Coding (BCC) which was a collaborative effort of Agere and Fraunhofer, as well as Parametric Stereo (PS) which was a research effort of Philips and Coding Technologies. The MPEG Surround efforts combined BCC and PS and recently became an international standard. Below we give a description of BCC and PS separately, and then we discuss how these technologies were combined into MPEG Surround.

6.1 Binaural Cue Coding (BCC) [21, 22]

BCC is based on the fact that humans localize sound in the azimuth plane, using the interaural level difference (ILD) and interaural time delay (ITD). ILD refers to the difference in level between the sound in the two eardrums, while ITD refers to the difference in time for the sound that reaches the two eardrums. The fact that humans need the sound in both ears to localize sound is mentioned as binaural detection. Thus, when listening to a 2-channel stereo recording of the same sound, it is possible to pan the "phantom" image of the sound from the center towards the left or right loudspeaker by introducing an appropriate amplitude attenuation and delay to the right or left loudspeaker respectively. Based on Lord Rayleigh's well-known Duplex Theory, ILD dominates in frequencies above 1.5 kHz, while ITD dominates in frequencies below 1.5 kHz. This is related to the wavelengths at these frequency ranges. Based on this fact, BCC attempts to create different sound sources in space using a monophonic recording (the sum signal), by correctly identifying and synthesizing the necessary cues for binaural detection. Consider now that that we have two sound sources at different directions. It is known that, even though in some areas of the time-frequency plane these sounds might overlap, humans can still correctly discriminate not only regarding the content but also regarding the direction of these two sources. This robustness of the human auditory system, which is able to group spectrally "similar" sounds in addition to using the ITD and ILD cues, also adds to the robustness of the BCC method. We also mention that due to the manner BCC operates, binaural unmasking is not encountered in BCC (or in PS), since the masking model is applied only once (in the downmixed signal) and all channels are based on the same spectral properties regarding the audio signal and the associated quantization noise.

BCC analyzes sound in the same manner as the human hearing system does, first analyzing the sounds in critical bands similarly to the human cochlea, and

then applying an envelope extraction method by rectifying and low-pass filtering the subband signals, similarly to the inner ear hair cells. Following this step, the BCC analysis system extracts in each critical band the interchannel level difference (ICLD) and the interchannel time difference (ICTD). These should not be confused with the ILD and ITD which are the differences between the sounds that reach the human eardrum. For playback using headphones, the interchannel differences are the same as the interaural differences. When using loudspeakers, these parameters differ, however it is assumed that if the interchannel differences are retained in the decoding, the final interaural cues will be the same as in the case of rendering the original multichannel recording.

The ICLD and ICTD are extracted in different subbands using a simple correlation analysis between a pair of channels, while the normalized cross-correlation between the two channels is also extracted (interchannel correlation - ICC). ICC is needed because it can control the width of the spatial sound image. These parameters are extracted for each audio segment, which is in the order of 30 ms according to binaural time resolution models.

Based on the preceding description of the BCC method, the basic 2 assumptions made by BCC are:

1. The ICLD, ICTD, and ICC per critical band are the only information needed in order to localize multiple sound sources in space. This is important, since binaural detection models have been verified experimentally for single sound sources, and it is therefore difficult to expect that for multiple sound sources (which might overlap in the time-frequency plane) the same models are valid.
2. The second assumption is that synthesizing the multiple audio channels from a monophonic source and the extracted binaural cues will result in recreating the auditory spatial image of the original multichannel recording.

These assumptions are essential to hold so that the BCC method can operate correctly. They are experimentally verified, based on listening tests which show that BCC indeed results in the correct spatial image.

For reducing the complexity of the analysis and synthesis modules, frequency transformations are based on the Fast Fourier Transform (FFT). For the analysis, the audio signals are divided in frames in the order of 30 ms using windowing (with 50% overlapping), and then each frame is transformed in the frequency domain using the FFT. The frequency samples are grouped into non-overlapping subbands, each having the bandwidth of 2 ERB (Equivalent Rectangular Bandwidth, 1 ERB is approximately equal to 1 Bark). For subband b, the binaural cues are extracted for channel c with respect to the reference channel (channel 1) based on the following relations:

$$ICLD_{c,b}(\text{dB}) = \lim_{l\to\infty} 10\log_{10}\left(\frac{\sum_{k=-l}^{l} x_c^2(k)}{\sum_{k=-l}^{l} x_1^2(k)}\right), \tag{5}$$

where $x_1(k)$ and $x_c(k)$ are the subband samples of two audio channels at time k.

$$ICTD_{c,b} = \arg\max_d \Phi_{1,c}(d), \tag{6}$$

where the normalized cross-correlation $\Phi_{1,c}(d)$ is given by

$$\Phi_{1,c}(d) = \lim_{l\to\infty} \frac{\sum_{k=-l}^{l} x_1(k)x_c(k+d)}{\sqrt{\sum_{k=-l}^{l} x_1^2(k) \sum_{k=-l}^{l} x_c^2(k)}}, \tag{7}$$

$$ICC_{c,b} = \max_d |\Phi_{1,c}(d)|. \tag{8}$$

The above relations are the definitions of the cues, based on which the cues are estimated in practice. As we can see, the binaural cues are extracted for each pair of channels. For multichannel signals, one channel is considered as the reference channel, and for all the remaining channels the binaural cues are extracted with respect to the reference channel. Finally, all channels are summed to create the downmix signal. The downmix signal is coded using a perceptual audio coder (*e.g.* MPEG Layer III), while the binaural cues are coded in the order of few kb/s as side information.

During synthesis, the inverse procedure is followed. The monophonic signal is analyzed in the frequency domain using the FFT (window of N samples), and the frequency coefficients of the sum signal S_n are organized in subbands and modified as follows for channel c and subband b:

$$S_{c,n} = F_{c,n} G_{c,n} S_n, \tag{9}$$

where $F_{c,n}$ is the factor determining the level difference and $G_{c,n}$ is the factor determining the phase difference for channel c with respect to the sum signal, for subband b. The index n refers to the frequency bins that correspond to subband b. The level difference is given from

$$F_{c,n} = 10^{(ICLD_{c,b}+r_{c,n})/10} F_{1,n}, \tag{10}$$

where $r_{c,n}$ is defined later, and $F_{1,n}$ is the level for the reference channel (channel 1), which in practice is a normalization term so that the power of the sum of all channels will be the same as the power of the sum signal (per subband), *i.e.,*

$$F_{1,n} = \frac{1}{\sqrt{1 + \sum_{i=2}^{C} 10^{(ICLD_{i,b}+r_{i,n})/10}}}. \tag{11}$$

The phase correction is given by

$$G_{c,n} = \exp\left(-j \frac{2\pi n (ICTD_{c,b} - \tau_b)}{N}\right), \tag{12}$$

where τ_b is the delay introduced in the reference channel. The term $r_{c,n}$ in the above equations is the only term in the BCC decoder which is affected by the ICC cues and is given by

$$r_{c,n} = (1 - ICC_{c,b})\hat{r}_{c,n}, \tag{13}$$

where $\hat{r}_{c,n}$ is a random sequence with uniform distribution. The term $r_{c,n}$ is used in order to reduce the correlation between the various channels for improving the final synthesized spatial image.

6.2 Parametric Stereo Coding (PS) [23]

Parametric Stereo (PS) is an alternative to BCC for Spatial Audio Coding. PS appeared later than BCC and consequently improves some of the concepts in BCC. In principle, though, PS is very similar to BCC, again extracting the binaural cues and using them as side information at the decoder.

The main steps of the PS encoder are:

- Analysis in audio segments using overlapping windows. The size of the window is around 23 ms following the minimal resolution of binaural hearing. However, for transient sounds the window size switches to a much smaller value (around 2 ms) in order to account for the precedence effect (explained previously) and also to avoid the pre-echo problem. The precedence effect in the case of transient signals must be examined because for such signals only the first 2 ms are enough to determine the spatial source of the sound.
- FFT or QMF (Quadrature Mirror Filters) filterbanks are applied for separation of the audio signals in critical bands (QMF being the of lower computational complexity). The bandwidth is determined equal to 1 ERB.
- Parameters extracted per subband: (i) the IID (interaural intensity difference, similarly to ICLD in BCC), (ii) the IPD (interaural phase difference) which is used instead of the ICTD in BCC, (iii) the interchannel coherence (IC) which is used similarly as ICC in BCC.
- Creation of the downmixed signal. In PS the downmix signal is not simply a sum signal as in BCC but is weighted in order to prevent phase cancellations and to ensure power preservation.
- Overall Phase Difference (OPD). The IPD extracted as explained above does not indicate which of the two channels (for a 2-channel stereo case) precedes the other, it only indicates the relative phase difference. Thus, an additional phase difference is extracted during encoding, the OPD, which is the phase difference between one of the channels (the reference channel) and the downmixed signal.
- The binaural cues (IID, IPD, IC, OPD) are coded based on perceptual criteria; in this chapter we describe the algorithm on a conceptual level and do not mention here the coding procedure.

The main steps of the PS decoder are:

- Signal decorrelation. In order to provide a more wide spatial image compared to BCC, the PS encoder artificially decorrelates the mono downmixed signal $s(k)$ in order to create a second signal for the decoder $s_d(k)$.
- From these two signals, a matrixed approach is followed in order to create the final two audio channels (for 2-channel stereo).

6.3 MPEG Surround Overview

As mentioned, in the standardization efforts of MPEG for Spatial Audio Coding (termed as MPEG Surround), a combination of BCC and PS has been adopted.

More specifically, in March 2004 the ISO/MPEG standardization group issued a "Call for Proposals" on Spatial Audio coding. From the four submissions that were received in response to that CfP, two were chosen based on extensive evaluations. These two were BCC (from Fraunhofer IIS and Agere) and PS (from Philips and Coding Technologies). MPEG standardization for MPEG Surround was completed on January 2007, and MPEG Surround is now an international standard (the documentation can be found as MPEG-D IS 23003-1). Below, the main components of MPEG surround are given.

Filterbank. A hybrid QMF filterbank was chosen, adopting the one used in PS.

OTT and TTT Elements. These are the levels where the binaural cues are extracted. The OTT (One-To-Two) element is the main part of the method, following BCC and PS on the extraction of the binaural cues from two-channel pairs of the original recording. The cues extracted are the Channel Level Difference (CLD) and the Interchannel Coherence/Cross-correlation (ICC). Time- and/or phase- differences are not extracted, due to the fact that they are not as important for recreating the correct spatial image. The TTT (Three-To-Two) elements are used for matrixing three channels into two, as explained next.

Hierarchical Encoding. The OTT and TTT elements can be combined in order to encode any N-channel recording to a M-channel encoded downmix (N>M). The OTT elements are applied to pairs of channels that have a front/back relation. For example, in a 5.1 setting the pairs encoded each by one OTT are the left and left surround, right and right surround, center and LFE. The result from these 3 OTT's will be 3 downmixed signals, which are in turn matrixed into 2 channels with one TTT. The TTT matrixing can be invertible or non-invertible. In the latter case, less parameters need to be coded and the third channel (Center axis) is estimated from the other two (Left and Right axis channels) using the CPC (Channel Prediction Coefficients) parameters extracted during encoding. However, in this non-invertible matrixing procedure, there is a loss of information and thus of spatial image.

Decorrelation. This part of MPEG Surround is also related with the decorrelation encountered in PS, where the matrixed signals are combined with their (artificially) decorrelated components, in order to provide a wider spatial image (*i.e.,* improving on the fact that the MPEG Surround encoding usually limits the spatial image of the initial multichannel recording).

Adaptive parameter smoothing. Especially in low bitrate applications, coarse quantization of the binaural cues might result in abrupt changes in the spatial image. Therefore, temporal smoothing of these parameters is suggested for smoother transitions of the spatial image.

Rates for the side information. Depending on the application and the available bandwidth (the MPEG Surround encoder is quality scalable), the rate for the side information can vary between 3 to 32 kb/s and above. The bitrate for the mono signal follows the available rates of mono or stereo encoding of MP3 or AAC encoding.

Residual Coding. For some applications transparency might be required. For such applications, the residual of the encoded signals might also be transmitted.

Backwards compatibility. The encoded signals are backwards compatible with MP3 and/or AAC decoders, and also with matrix decoders (such as Dolby Pro Logic II).

Artistic dowmnix capability. An issue that is recognized and addressed in MPEG Surround, is the fact that for many multichannel recordings, a 2-channel stereo recording is also available (usually in new consumer audio formats such as DVD-Audio and SACD). This stereo recording is referred to as the artistic downmix, which exactly recognizes the fact that it might be created with a different mixing procedure (usually depending on artistic criteria) compared to the creation of the multichannel mix. In this case, the downmixed stereo by SAC might be different than the available artistic downmix. In this case, either the SAC downmix or the artistic downmix will be transmitted. In the former case, the multichannel upmix (*i.e.,* the result of the SAC decoder) will be correct (*i.e.,* similar spatial image to the original multichannel recording); however, for those consumers relying on the stereo downmix (*i.e.,* if the SAC decoder is not available) the stereo dowmix will not sound similar to the artistic downmix. On the other hand, if the artistic downmix is transmitted along with the SAC cues, the SAC decoder will not "correctly" upmix the stereo signal into the multichannel recording. The solution proposed is to transmit the artistic downmix, along with additional parameters which will convert the artistic downmix closer to the SAC downmix. The latter can then be correctly decoded by the SAC decoder if available.

Ongoing work. Ongoing research within the MPEG Surround framework has concentrated mainly on the issue of non-guided decoding. This concept is very similar to stereo upmixing. Given *e.g.,* a 2-channel stereo recording, the question is how can it be upmixed so that *e.g.,* it can be reproduced as a 5.1 multichannel recording. The assumption is that the binaural cues for the multiple channels, which are now missing, could be extracted from the available binaural cues of the stereo recording. Some initial experiments based on this concept have shown significant improvement compared to matrix-based upmixing.

7 Source/Filter Model for Immersive Audio [24]

At a point where MPEG Surround (explained in the previous paragraphs) achieves coding rates for 5.1 multichannel audio that are similar to MP3 coding rates for 2-channel stereo, it seems that the research in audio coding might have no future. However, this is far from the truth. On the opposite, current multichannel audio formats will eventually be substituted by more advanced formats which will allow for truly immersive audio environments. Future audiovisual systems will not distinguish between whether the user will be *watching* a movie or *listening* to a music recording; audiovisual reproduction systems of the future are envisioned to offer a realistic experience to the consumer who will be *immersed* into the audiovisual content. As opposed to *listening* and *watching*, the passive voice of *immersed* implies that the user's environment will be seamlessly transformed into the environment of his desire, which in turn implies that the user is in fact not a passive receiver of the content but can interact with the content according to his will. It is important to note the fact that using a large number of loudspeakers is useless if there is no increase in the content information. Immersive audio is largely based on enhanced audio content, which translates into using a large number of microphones for obtaining a recording containing as many as possible sound sources. These sources offer increased sound directions around the listener during reproduction, but are also useful for providing interactivity between the user and the audio environment. The increase in audio content combined with the strict requirement regarding the processing and network delays and losses in the coding and transmission of immersive audio content are the issues that are addressed by the methods described in this and the following sections.

Before proceeding to describing the proposed source/filter and sinusoidal models, it is necessary to briefly explain how interactivity can be achieved using the multiple microphone recordings (microphone signals) of a particular multichannel recording. The number of these multiple microphone signals is usually higher than the available loudspeakers, thus a mixing process is needed when producing a multichannel audio recording. Mixing of the multi-microphone audio recordings at the decoder side is considered in this last part of this chapter. Remote mixing is imperative for almost all immersive audio applications, since it offers the amount of freedom for the creation of the content that is needed for interactivity. On the other hand, remote mixing implies that an even higher number of audio channels must be stored at or transmitted to the consumer side. Thus, a key difference of immersive audio compared to multichannel audio is the increased demand for transmission rates due to the needed interactivity. It is of great importance to explain that in an immersive audio application, multichannel methods such as MPEG Surround cannot be applied. This is due to the fact that, for achieving interactivity through remote mixing, not only the spatial image (as in MPEG Surround) but the exact content of each microphone recording must be retained by the coding method.

In this and the following sections we describe how the source/filter and the sinusoidal models can be applied for low bitrate immersive audio coding. These

models have been very popular for modeling speech signals [25], but in audio coding so far they have been found to degrade the audio quality, especially in low bitrate coding where the number of parameters must remain low. Regarding a description of previous efforts on applying the source/filter and sinusoidal models in low bitrate audio coding the reader is referred to [6] (in the following we also indicate some representative previous methods). In this section, we describe a recently proposed method for applying the source/filter model for modeling spot microphone recordings of multichannel audio. These are the recordings that are obtained for multichannel audio applications, before the mixing process. A large number of microphones in a venue is used, to create a multichannel audio recording. These are then mixed in order to produce the final multichannel audio recording. It would be desirable to transmit the multiple microphone signals of a performance, before those are mixed into the (usually much smaller number of) channels of the multichannel recording. As explained, this would allow for interactive applications that are of immense interest for immersive audio environments, such as remote mixing of the multichannel recording and remote collaboration of geographically distributed musicians [26]. For these applications, the number of audio channels to be encoded is higher than in multichannel recordings, and low bitrate encoding for each channel is important.

In this section, the source/filter model is applied to multichannel audio spot signals, with the final objective of encoding the multiple microphone signals of a music performance with moderate datarate requirements. This would allow for transmission through low bandwidth channels such as the current Internet infrastructure or wireless channels. The method focuses on the microphone signals of a performance *before they are mixed*, and thus can be applied to immersive applications such as remote mixing and distributed performances. In principle, this method attempts to model each microphone signal with respect to a reference audio signal, so in this sense it follows the Spatial Audio Coding (SAC) philosophy. However, while in SAC the objective is to retain the *spatial image* of the original (before the coding stage) multichannel recording, the objective in the method described in this section is to retain the *content* of each of the spot microphone signals. In both cases, audio quality is of central importance.

It is of interest to mention that source/filter models for audio coding have been proposed previously, *e.g.,* [6]. In most approaches the focus is on modeling the excitation part of the signal (residual error), which justifies our approach on obtaining the excitation signal from the reference channel, described in the following sections. The other issue in source/filter models for audio coding is improving the estimation of the spectral envelope compared to conventional LPC, which in our method is achieved by a multiresolution approach. For low bitrate coding, the TwinVQ [27] (Transform-domain Weighted Interleave Vector Quantization) has been implemented as part of MPEG-4 audio coding activities (scalable audio coding). However, TwinVQ at low rates results in degradation of the audio quality (around 3.0 score below 16 kb/s).

7.1 Spot Microphone Signals

A brief description is given below, of how the multiple microphone signals for multichannel rendering are recorded. The focus is mainly on live concert hall performances, however there is no loss of generality for the proposed methods. A number of microphones is used to capture several characteristics of the venue, resulting in an equal number of microphone signals (stem recordings). The goal is to design a system based on available microphone signals, that is able to recreate all of these target microphone signals from a smaller set (or even only one, which could be a sum signal) of reference microphone signals at the receiving end. The result would be a significant reduction in transmission requirements, while enabling interactivity at the receiver. By examining the acoustical characteristics of the various stem recordings, the distinction of microphones is made into reverberant and spot microphones.

Spot microphones are microphones that are placed close to the sound source. The recordings of these microphones heavily depend on the instruments that are near the microphone and not so much on the hall acoustics; these recordings recreate the sense that the sound source is not a point source but rather distributed such as in an orchestra. Resynthesizing the signals captured by these microphones, therefore, involves enhancing certain instruments and diminishing others, which in most cases overlap in the time and frequency domains. Reverberant microphones are the microphones placed far from the sound source, that mainly capture the reverberation information of the venue. Here, the recordings made by spot microphones are considered, since modeling their spectral properties is more challenging compared to reverberant microphone signals. Modeling of the latter signals has been considered in earlier work, where linear time-invariant filters were proposed for transforming a reference signal into a given reverberant signal [28].

7.2 Model and Motivation

The proposed methodology, which is based on a multiband source / filter representation of the multiple microphone signals, consists of the following steps. Each microphone signal is segmented into a series of short-time overlapping frames. For each frame, the audio signal is considered approximately stationary, and the spectral envelope is modeled as a vector of linear predictive (LP) coefficients [14]. Under the source/filter model, the signal $s(n)$ at time n is related with the p previous signal samples by the following autoregressive (AR) equation

$$s(n) = \sum_{i=1}^{p} a(i)s(n-i) + e(n), \tag{14}$$

where $e(n)$ is the modeling error (residual signal), and p is the AR filter order. In the frequency domain, this relation can be written as

$$P_s(\omega) = |A(\omega)|^{-2} P_e(\omega), \tag{15}$$

where $P_x(\omega)$ denotes the power spectrum of signal $x(n)$. $A(\omega)$ denotes the frequency response of the AR filter, *i.e.,*

$$A(\omega) = 1 - \sum_{i=1}^{p} a(i)e^{-j\omega i}. \tag{16}$$

The $p+1^{th}$-dimensional vector $\boldsymbol{a}^T = [1,\ -a_1,\ -a_2,\ \cdots,\ -a_p]^T$ is the low dimensional representation of the signal spectral properties. If $s(n)$ is an AR process, the noise $e(n)$ is white, thus $\boldsymbol{a}$ completely characterizes the signal spectral properties. In the general case, the error signal will not have white noise statistics and thus cannot be ignored. In this general case, the all-pole model that results from the LP analysis gives only an approximation of the signal spectrum, and more specifically the spectral envelope. For the particular case of audio signals, the spectrum contains only frequency components that correspond to the fundamental frequencies of the recorded instruments, and all their harmonics. The AR filter for an audio frame will capture its spectral envelope. The error signal is the result of the audio frame filtered with the inverse of its spectral envelope. Thus, we conclude that the error signal will contain the same harmonics as the audio frame, but their amplitudes will now have significantly flatter shape in the frequency spectrum.

Consider now two microphone signals of the same music performance, captured by microphones placed close to two different groups of instruments of the orchestra. Each of these microphones mainly captures that particular group of instruments, but also captures all the other instruments of the orchestra. For simplification, consider that the orchestra consists of only two instruments, *e.g.,* a violin and a trumpet. Microphone 1 is placed close to the violin and microphone 2 close to the trumpet. It is true in most practical situations, that microphone 1 will also capture the trumpet, in much lower amplitude than the violin, and vice versa for microphone 2. In that case, the signal s_1 from microphone 1, and the signal s_2 from microphone 2 will contain the fundamentals and corresponding harmonics of both instruments, but they will differ in their spectral amplitudes. Consider a particular short-time frame for these 2 signals, which corresponds to the exact same music part (*i.e.,* some time-alignment procedure will be necessary to align the two microphone signals). Each of the two audio frames is modeled with the source/filter model:

$$s_k(n) = \sum_{i=1}^{p} a_k(i)s_k(n-i) + e_k(n),\ k = 1, 2. \tag{17}$$

From the previous discussion it follows that the two residual signals e_1 and e_2 will contain the same harmonic frequency components. If the envelope modeling was perfect, then it follows that they would also be equal (differences in total gain are of no interest for this application), since they would have flat magnitude with exactly the same frequency components. In that case, it would be possible to resynthesize each of the two audio frames using only the AR filter that corresponds to that audio frame, and the residual signal of the other microphone. If this model was used similarly for all the spot microphone signals of a single performance, it would be possible to completely resynthesize these signals using their AR vector sequences (one vector for each audio frame) and the residual

error of only one microphone signal. This would result in a great reduction of the data rate of the multiple microphone signals.

In practice, the AR filter is not an exact representation of the spectral envelope of the audio frame, and the residual signals for the two microphone signals will not be equal. However, the modeling performance of the AR filter can be improve by using filterbanks. The spectrum of the audio signals is divided in subbands and LP analysis is applied in each band separately (subband signals are downsampled). A small AR filter order for each band can result in much better estimation of the spectral envelope than a high-order filter for the full frequency band. The multiband source/filter model achieves a flatter frequency response for the residual signals. Then, one of them can be used for resynthesizing the other microphone signals, in the manner explained in the previous paragraph. However, the error signals cannot be made exactly equal, thus the resynthesized signals will not sound exactly the same as the originally recorded signals. This has been found to result in crosstalk between the modeled spot signals, however the audio quality remains high. In other words, the "main" group of instruments that is captured still remains the prominent part of the microphone signal, while other parts of the orchestra might be more audible in the resynthesized signal than in the original microphone signal. Returning to the example of the two microphones and the two instruments, if the residual of microphone 1 is used to resynthesize the signal of microphone 2, then in the result the violin will most likely be more audible than in the original microphone 2 signal. This happens because some information of the first microphone signal remains in the error signal, since the spectral envelope modeling is not perfect. However, the trumpet will still be the prominent of the two instruments in the resynthesized signal for mic 2, since we used the original spectral information of that microphone signal.

These claims hold for any type of harmonic signals, *e.g.,* speech signals. Some types of microphone signals, such as percussive signals and signals from reverberant microphones, present different challenges. Especially for sounds such as percussive sounds which cannot be accurately modeled by their spectral envelope only, the sinusoidal model can alternatively be used as described next.

7.3 Implications

So far, we have described how the source/filter model can be applied to spot microphone signals modeling. From the above discussion, it is clear that the method consists of coding one audio signal only (reference channel), which can be a downmix of all the spot recordings, along with side information consisting of the subband LPC envelopes of all the short-time frames for all microphone signals. It is important to show what bitrates are needed for this side information, and the achieved quality of both the modeling and the coding procedures. Since coding of LPC envelopes is a problem that has been treated extensively for speech signals, the details are not given here and can be found in [24]. The results of this work indicate a high audio quality both for modeling and coding (subjective scores around 4.0 compared to the original recording) for bitrates as low as 5 kb/s for the side information of each spot signal. We mention again that crosstalk

is introduced to the modeled signals, and the importance of this fact depends on the particular application and is an issue under investigation. Alternatively, the sinusoidal model (examined next) can be employed for alleviating the crosstalk problem, at the expense of the need for higher bitrates for coding.

8 Sinusoidal Model for Immersive Audio [29]

As mentioned in the previous section, the sinusoidal model when applied for low bitrate audio coding, has been found to degrade audio quality. In this section, we describe how this model can be applied to spot microphone signals and result in good audio quality. The sinusoids plus noise model is employed for each spot microphone signal, and models each signal with the sinusoidal parameters (harmonic part) and the short-time spectral envelope of the noise (modeling noise part). For resynthesis of each microphone signal the harmonic part that was fully encoded is added to the noise part, which is recreated by using the signal's corresponding noise envelope with the noise residual obtained from the reference signal. This procedure, which is termed in [29] as *noise transplantation*, is based on the observation that the noise signals of the various spot signals of the same multichannel recording are very similar when the harmonic part has been captured with a high enough number of sinusoids. The sinusoids plus noise model has been applied to audio (music) signals under various approaches (more in Section 8.1). To our knowledge, this is the first attempt to apply (and tailor) this model to the specific case of immersive audio, for low bitrate immersive audio coding.

8.1 Sinusoids Plus Noise Model

The sinusoidal model represents a harmonic signal $s(n)$ as the sum of a small number of sinusoids with time-varying amplitudes and frequencies

$$s(n) = \sum_{l=1}^{L} A_l(n)\, cos(\theta_l(n)) \tag{18}$$

To find the parameters of the model, one needs to segment the signal into a number of short-time frames and compute the short-time Fourier transform (STFT) for each frame, and then identify the prominent spectral peaks using a peak detection algorithm. Each peak is represented as a triad of the form $(A_l^q, \omega_l^q, \varphi_l^q)$ (amplitude, frequency, phase), which corresponds to the l^{th} sinewave of the q^{th} frame. A peak continuation algorithm is needed in order to assign each peak to a frequency trajectory by matching the peaks of the previous frame to the current frame, using linear amplitude interpolation and cubic phase interpolation.

Several variations of the sinusoids plus noise model have been proposed for applications such as signal modifications and low bitrate coding, focusing on three different problems: (1) accurately estimating the sinusoidal parameters

from the original spectrum, (2) representing the modeling error (noise component), and (3) representing signal transients. Problem (1) has been extensively treated for speech signals, *e.g.* [30, 31], and variations of these approaches have been extended to wideband audio. For addressing problem (3) use of damped sinusoids and AM modulated sinusoids (instead of constant amplitude sinusoids) have been proposed (*e.g.* [32, 33]). Here, as explained, we focus on the problem of noise representation. In music, a harmonic plus noise model was first proposed in [34], where the noise part was modeled based on a piecewise-linear approximation of its short-time spectral envelope or alternatively its LPC envelope (assuming white noise excitation during synthesis). More recent is the work in [35], where multiresolution analysis was applied for better estimating the sinusoidal parameters by passing the signal through an octave-spaced filterbank which was designed for avoiding aliasing problems. Regarding the noise part, it was not parametrically modeled for best audio quality. The work in [36] and more recently [37] has focused on the noise part modeling. In the former approach, the noise energy at each critical band was only retained, forming a perceptual spectral envelope of the noise signal. In the latter, the perceptual envelope is estimated based on a perceptually motivated LPC estimation. Thus, more recent methods focus on modeling the noise using only its (short-time) perceptually relevant spectral envelope. While these methods offer the advantage of low bitrate coding for the noise part, the resulting audio quality is usually worse than the quality of the original audio signal (subjective results with average grades around 3.0 in a 5-grade scale have been reported). Within the MPEG-4 activities, two methods were proposed for sinusoidal audio coding, namely ASAC [38] (Analysis/Synthesis Audio Codec) and HILN (Harmonic and Individual Lines plus Noise) [39] (the latter was included in MPEG-4 as the recommended low-bitrate parametric audio coder). Both have been developed for low-bitrate audio coding and audio quality at low rates deteriorates significantly (subjective quality below 3.0 for rates below 16 kb/s).

Here, the interest is on high-quality low-bitrate audio modeling (achieving a grade around 4.0 is desirable). Regarding the modeling procedure, any sinusoidal modeling algorithm can be applied. The focus here is on modeling the noise part of the sinusoidal model. It is important to derive a method which results in good audio quality compared not only to the sinusoids-only model but also compared to the original recording.

The sound representation is obtained by restricting the sinusoids to modeling only the deterministic part of the sound, leaving the rest of the spectral information in the noise component $e(n)$, *i.e.,* for each short-time frame the signal can be represented as

$$s(n) = \sum_{l=1}^{L} A_l(n)\, cos(\theta_l(n)) + e(n). \tag{19}$$

After the sinusoidal parameters are estimated, the noise component is computed by subtracting the harmonic component from the original signal. In this section, the noise component of the sinusoidal is modeled as the result of filtering a

residual noise component with an autoregressive (AR) filter that models the noise spectral envelope. Linear Predictive (LP) analysis is applied to estimate the spectral envelope of the sinusoidal noise. In other words, the assumption is that the following equation for the noise component of the sinusoidal model holds:

$$e(n) = \sum_{i=1}^{p} \alpha(i)\, e(n-i) + r_e(n). \tag{20}$$

The quantity $e(n)$ is the sinusoidal noise component, while $r_e(n)$ is the residual of the noise, and p is the AR filter order. The $p+1^{th}$-dimensional vector $\boldsymbol{\alpha}^T = [1,\ -\alpha_1,\ -\alpha_2,\ \cdots,\ -\alpha_p]^T$ represents the spectral envelope of the noise component $e(n)$. In the frequency domain (20) becomes

$$S_e(e^{j\omega}) = \Big|\frac{1}{A(e^{j\omega})}\Big|^2 S_{r_e}(e^{j\omega}), \tag{21}$$

where $S_e(e^{j\omega})$ and $S_{r_e}(e^{j\omega})$ is the power spectrum of $e(n)$ and $r_e(n)$, respectively, while $A(e^{j\omega})$ is the frequency response of the LP filter $\boldsymbol{\alpha}$. Since in this section there are two noise quantities introduced, *i.e.,* the sinusoidal model noise e, and its whitened version r_e, we will refer to e as the (sinusoidal) *noise* signal, and to r_e as the *residual* (noise) of e. For convenience, we refer to the Sinusoids plus Noise Model as SNM, which in practice can be any implementation of sinusoidal modeling.

8.2 Noise Transplantation

Consider two spot microphone signals of a music performance, in which the two microphones are placed close to two distinct groups of instruments of the orchestra. The first microphone signal is denoted by $x_L(n)$ (for simplicity we refer to this signal as the left channel, which should not be confused with the channels of the multichannel mix), while the second one is denoted by $x_R(n)$ (referred to as the right channel). Each of these microphone signals mainly captures the sound from the closest group of instruments, but also captures the sound from all the other instruments of the orchestra (this is especially true for live concert hall performances). Thus, the two recordings are similar in content, and this is apparent in most multichannel recordings in such settings. Alternatively, one of the channels (the reference signal) could be a sum signal of all the spot recordings.

Sinusoidal models capture the harmonics of the original audio signal well if the number of harmonics used is carefully chosen. However, especially for music signals, the harmonic component is not sufficient for high-quality synthesis; its structured nature and the lack of "randomness" in the signal is audible even if a high number of sinusoids is used. The noise signal e, which contains the spectral information which is considered of random nature, is necessary for high-quality audio synthesis. It mostly contains higher-frequency information, and adds the acoustically needed "randomness" to the sinusoidal component. In coding applications, the noise signal will require a much higher degree in terms of datarates

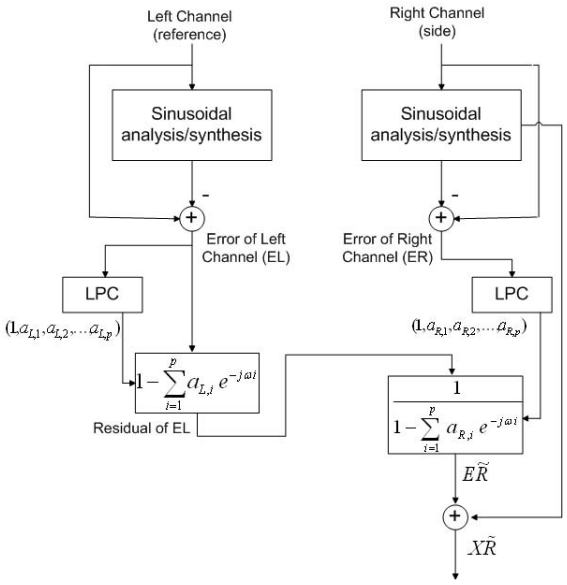

Fig. 7. Noise transplantation. The LPC residual of the reference signal's noise component is filtered by the side signal's noise envelope and added to its sinusoidal component.

compared to the sinusoidal component, exactly due to its quasi-random nature. Thus, here a model is described that is based on the sinusoidal component of the audio signal, but can result in high-quality audio synthesis at the decoder. In order to achieve this objective, the proposed scheme is similar to the Spatial Audio Coding philosophy. In other words, given a collection of microphone signals that correspond to the same multichannel recording (and thus have similar content), only one of them is encoded as a full audio channel (reference signal). We model the remaining signals with the SNM model, retaining their sinusoidal components and the noise spectral envelope (filter $\boldsymbol{\alpha}$ in (20)). For resynthesis, the reference signal is modeled with the SNM in order to obtain its noise signal e, and from it the LP residual r_e is obtained using LPC analysis. Finally, each microphone signal is reconstructed using its sinusoidal component and its noise LP filter; its sinusoidal component is added to the noise component that is obtained by filtering (with the signal's LP noise shaping filter) the LPC residual of the sinusoidal noise from the reference signal. The assumption is that, as the harmonics capture most of the important information for each microphone signal, the noise part that remains will be similar for all the microphone signals of the same multichannel recording. This assumption is verified in recent results [29]. By taking the reference residual (whitened sinusoidal noise) and filtering it with the correct noise envelope (the envelope of side channel k, where the reference and side signals must be time-aligned), a noise signal is obtained with very similar spectral properties to the initial noise component of the side channel k. This procedure is depicted in the diagram of Fig. 7.

To formalize the previous discussion, considering a multichannel recording with M microphone signals, the relation for the resynthesis of one of the *side* microphone signals x_k (as opposed to the *reference* signal $x_{(ref)}$) is

$$\hat{x}_k(n) = \sum_{l=1}^{L} A_{k,l}(n)\, cos(\theta_{k,l}(n)) + \hat{e}_k(n)\ , k = 1, ..., M, \tag{22}$$

where $\hat{e}_k(n)$ is represented in the frequency domain as

$$\hat{S}_{e_k}(e^{j\omega}) = \left| \frac{1}{1 - \sum_{i=1}^{p} \alpha_k(i)\, e^{-j\omega i}} \right|^2 S_{r_{e_{(ref)}}}(e^{j\omega}). \tag{23}$$

In the equations above, $A_{k,l}(t)$ and $\theta_{k,l}(t)$ are the estimated sinusoidal parameters of microphone signal k and α_k the signal's LP noise shaping filter, while $\hat{e}_k(n)$ is the estimated noise component using the noise transplantation procedure described, *i.e.* filtering, with the noise shaping filter α_k, of the reference signal residual noise. The residual of the noise component of the reference signal can be found as

$$S_{r_{e_{(ref)}}}(e^{j\omega}) = \left| 1 - \sum_{i=1}^{p} \alpha_{(ref)}(i)\, e^{-j\omega i} \right|^2 S_{e_{(ref)}}(e^{j\omega}). \tag{24}$$

Thus, $S_{r_{e_{(ref)}}}(e^{j\omega})$ is the power spectrum of the reference signal noise residual (AR modeling error of the sinusoidal noise), and $e_{(ref)}$ is the sinusoidal noise obtained from the reference. For the reference signal, the SNM model is applied only for obtaining the noise residual. This signal is assumed to be encoded and transmitted as a monophonic audio signal (*e.g.* MP3) to the receiver. Also, it is possible that more than one reference signals might be necessary for the method to perform well in practice, depending on the nature of the multiple signals of a particular multichannel recording or when backwards compatibility with stereo decoders is required.

8.3 Implications

As in the case of the source/filter model, in the noise transplantation procedure for sinusoidal modeling of the spot microphone signals the objective is to transmit only one audio channel and small side information for the remaining signals. For each spot signal (except from the reference channel), the sinusoidal parameters and the LPC envelopes of the noise must be transmitted. For the LPC envelopes, based on our experience of the source/filter model, we can expect datarates in the order of 5 kb/s for high audio quality. Regarding the sinusoidal parameters, experiments are currently underway. It is known from previous work on coding of the sinusoidal parameters *e.g.* [40] that around 20 bits per sinusoid are needed for good quality. In [29] it is shown that a 4.0 grade can be achieved in modeling using the noise transplantation procedure using 30 sinusoids per signal frame (updated every 10 ms). Thus, we can expect datarates in the order

of 60 kb/s, for good quality coding, however our efforts have been concentrated for rates below 30 kb/s. In any case, we mention that the source/filter and the sinusoidal modeling methods that were described are still under investigation regarding the achieved bitrates and the quality that can be obtained.

9 Conclusions

In this chapter, a review was given of multichannel audio coding methods, from monophonic and stereophonic coding to multichannel audio coding. We placed interest on multichannel audio coding methods for low bitrate coding applications which are currently under development and recently became an International Standard under the name of MPEG Surround. Finally, the source/filter and sinusoidal models for multichannel and immersive audio were presented, and shown to offer high quality audio for low bitrate applications. These models offer the advantage of a parametric representation of the audio signals, and in addition to low bitrate coding the model parameters can be estimated using statistical estimation methods which is important in the case *e.g.* of packet losses [41]. At the same time, by focusing on spot microphone signals (before the mixing process), interactivity between the user and the auditory image becomes possible, which is essential for truly immersive intelligent environments.

10 Resources

An indicative resource list follows, related with the matterial covered in this chapter.

- A document by the Audio Engineering Society, related to the principles of surround 5.1 sound:
  http://www.aes.org/technical/documents/AESTD1001.pdf
- MPEG-1 and MPEG-2 Audio links:
  http://www.mpeg.org/MPEG/audio.html
  http://www.chiariglione.org/mpeg/
  http://www.iis.fraunhofer.de/EN/bf/amm/index.jsp
- MPEG-1 and MPEG-2 Audio source code:
  ftp://ftp.tnt.uni-hannover.de/pub/MPEG/audio/
  http://sourceforge.net/projects/faac/
- Dolby matrixing technologies:
  http://www.dolby.com/
- Official information about MPEG Surround:
  http://www.mpegsurround.com/
- The Integrated Media Systems Center (IMSC) of the University of Southern California (USC) has pioneered the research in Immersive Audio technologies:
  http://imsc.usc.edu/

References

1. ITU-R BS.1116, "Methods for the subjective assessment of small impairments in audio systems including multichannel sound systems," 1994. International Telecommunications Union, Geneva, Switzerland.
2. ISO/IEC JTC1/SC29/WG11 (MPEG) International Standard ISO/IEC 11172-3, "Coding of moving pictures and associated audio for digital storage media at up to about 1.5 Mbit/s," 1992.
3. D. Pan, "A tutorial on MPEG/Audio compression," *IEEE Multimedia*, pp. 60–74, Summer 1995.
4. P. Noll, "MPEG digital audio coding," *IEEE Signal Processing Magazine*, pp. 59–81, September 1997.
5. K. Brandenburg, "MP3 and AAC explained," *in Proc.* 17^{th} *International Conference on High Quality Audio Coding of the Audio Engineering Society (AES)*, September 1999.
6. T. Painter and A. Spanias, "Perceptual coding of digital audio," *Proc. IEEE*, vol. 88, pp. 100–120, April 2000.
7. H. S. Malvar, "Lapped transforms for efficient transform/subband coding," *IEEE Trans. Acoust., Speech, and Signal Process.*, vol. 38, pp. 969–978, June 1990.
8. ISO/IEC JTC1/SC29/WG11 (MPEG) International Standard ISO/IEC 13818-3, "Generic coding of moving pictures and associated audio: Audio," 1994.
9. ISO/IEC JTC1/SC29/WG11 (MPEG) International Standard ISO/IEC 13818-7, "Generic coding of moving pictures and associated audio: Advanced audio coding," 1997.
10. M. Bosi, K. Brandenburg, S. Quackenbush, L. Fielder, K. Akagiri, H. Fuchs, M. Dietz, J. Herre, G. Davidson, and Y. Oikawa, "ISO/IEC MPEG-2 advanced audio coding," *in Proc.* 101^{st} *Convention of the Audio Engineering Society (AES), preprint No. 4382*, (Los Angeles, CA), November 1996.
11. K. Brandenburg and M. Bosi, "ISO/IEC MPEG-2 Advanced Audio Coding: Overview and applications," *in Proc.* 103^{rd} *Convention of the Audio Engineering Society (AES), preprint No. 4641*, 1997.
12. J. D. Johnston and A. J. Ferreira, "Sum-difference stereo transform coding," in *IEEE Int. Conf. Acoust., Speech, Signal Process. (ICASSP)*, pp. 569–572, 1992.
13. B. C. J. Moore, *An Introduction in the Psychology of Hearing*. Academic Press, 1989.
14. S. Haykin, *Adaptive Filter Theory*. Prentice Hall, 1996.
15. J. Herre, K. Brandenburg, and D. Lederer, "Intensity stereo coding," in *Proc.* 96^{th} *Convention of the Audio Engineering Society (AES), preprint No. 3799*, February 1994.
16. R. Dressler, "Dolby Surround Pro Logic decoder principles of operation." http://www.dolby.com.
17. "Dolby Surround Pro Logic II decoder principles of operation." http://www.dolby.com.
18. D. Yang, H. Ai, C. Kyriakakis, and C. J. Kuo, "High-fidelity multichannel audio coding with karhunen-loeve transform," *IEEE Trans. on Speech and Audio Processing*, vol. 11, pp. 365–380, July 2003.
19. R. Irwan and R. M. Aarts, "Two-to-five channel sound processing," *J. Audio Eng. Soc.*, vol. 50, pp. 914–926, November 2002.
20. J. Breebaart, J. Herre, C. Faller, J. Roden, F. Myburg, S. Disch, H. Purnhagen, G. Hotho, M. Neusinger, K. Kjorling, and W. Oomen, "MPEG Spatial Audio

Coding / MPEG Surround: Overview and current status," in *Proc. AES 119th Convention, Paper 6599*, (New York, NY), October 2005.
21. F. Baumgarte and C. Faller, "Binaural cue coding - Part I: Psychoacoustic fundamentals and design principles," *IEEE Trans. Speech and Audio Process.*, vol. 11, pp. 509–519, November 2003.
22. C. Faller and F. Baumgarte, "Binaural cue coding - Part II: Schemes and applications," *IEEE Trans. Speech and Audio Process.*, vol. 11, pp. 520–531, November 2003.
23. J. Breebaart, S. van de Par, A. Kohlrausch, and E. Schuijers, "Parametric coding of stereo audio," *EURASIP Journal on Applied Signal Processing*, pp. 1305–1322, 2005:9.
24. K. Karadimou, A. Mouchtaris, and P. Tsakalides, "Multichannel audio modeling and coding using a multiband source/filter model," *in Proc. 39nd Annual Asilomar Conference on Signals, Systems and Computers*, (Pacific Grove, CA), October 30 - November 2, 2005.
25. L. Rabiner and B.-H. Juang, *Fundamentals of Speech Recognition*. Prentice Hall, 1993.
26. A. Sawchuk, E. Chew, R. Zimmermann, C. Papadopoulos, and C. Kyriakakis, "From remote media immersion to distributed immersive performance," *in Proc. ACM SIGMM Workshop on Experiential Telepresence (ETP)*, (Berkeley, CA), November 2003.
27. N. Iwakami, T. Moriya, and S. Miki, "High-quality audio coding at less than 64 kbit/s by using transform-domain weighted interleave vector quantization (TWINVQ)," in *IEEE Int. Conf. Acoust., Speech, Signal Process. (ICASSP)*, pp. 3095–3098, May 1995.
28. A. Mouchtaris, S. S. Narayanan, and C. Kyriakakis, "Virtual microphones for multichannel audio resynthesis," *EURASIP Journal on Applied Signal Processing, Special Issue on Digital Audio for Multimedia Communications*, vol. 2003:10, pp. 968–979, 2003.
29. C. Tzagkarakis, A. Mouchtaris, and P. Tsakalides, "Sinusoidal modeling of spot microphone signals based on noise transplantation fro multichannel audio coding." Submitted *European Signal Process. Conf. (EUSIPCO)*, 2007.
30. R. J. McAulay and T. F. Quatieri, "Speech analysis/synthesis based on a sinusoidal representation," *IEEE Trans. Acoust., Speech, and Signal Process.*, vol. 34(4), pp. 744–754, August 1986.
31. Y. Stylianou, "Applying the harmonic plus noise model in concatinative speech synthesis," *IEEE Trans. Speech and Audio Process.*, vol. 9(1), pp. 21–29, 2001.
32. J. Jensen, R. Heusdens, and S. H. Jensen, "A perceptual subspace approach for modeling of speech and audio signals with damped sinusoids," *IEEE Trans. Speech and Audio Process.*, vol. 12, no. 2, pp. 121–132, 2004.
33. M. G. Christensen, A. Jakobsson, S. V. Andersen, and S. H. Jensen, "Linear AM decomposition for sinusoidal audio coding," in *IEEE Int. Conf. Acoust., Speech, Signal Process. (ICASSP)*, pp. 165–168, 2005.
34. X. Serra and J. O. Smith, "Spectral modeling sythesis: A sound analysis/synthesis system based on a deterministic plus stochastic decomposition," *Computer Music Journal*, vol. 14(4), pp. 12–24, Winter 1990.
35. S. N. Levine, T. S. Verma, and J. O. Smith, "Multiresolution sinusoidal modeling for wideband audio with modifications," *IEEE Int. Conf. Acoust., Speech, Signal Process. (ICASSP)*, 1998.
36. M. Goodwin, "Residual modeling in music analysis-synthesis," in *IEEE Int. Conf. Acoust., Speech, Signal Process. (ICASSP)*, May 1996.

37. R. C. Hendriks, R. Heusdens, and J. Jensen, "Perceptual linear predictive noise modelling for sinusoid-plus-noise audio coding," in *IEEE Int. Conf. Acoust., Speech, Signal Process. (ICASSP)*, May 2004.
38. B. Edler, H. Purnhagen, and C. Ferekidis, "ASAC - analysis/synthesis audio codec for very low bit rates," *in Proc.* 100^{th} *Convention of the Audio Engineering Society (AES), Preprint No. 4179*, May 1996.
39. H. Purnhagen and N. Meine, "HILN - the MPEG-4 parametric audio coding tools," in *IEEE International Symposium on Circuits and Systems (ISCAS)*, pp. 201–204, May 2000.
40. R. Vafin and W. B. Kleijn, "On frquency quantization in sinusoidal audio coding," *IEEE Signal Processing Letters*, vol. 12, no. 3, pp. 210–213, 2005.
41. K. Karadimou, A. Mouchtaris, and P. Tsakalides, "Packet loss concealment for multichannel audio using the multiband source/filter model," in *Conf. Record of the Asilomar Conf. Signals, Systems and Computers*, (Pacific Grove, CA), November 2006.

View publication stats