Efficient Coding of Natural Sounds

Von der Fakultät für Mathematik und Naturwissenschaften

der Carl-von-Ossietzky-Universität Oldenburg

zur Erlangung des Grades und Titels eines

Doktors der Naturwissenschaften (Dr. rer. nat.)

angenommene Dissertation

Dipl.-Math. M.Sc. Stefan Strahl

geboren am 6. März 1975

in Augsburg

Gutachter: Prof. Dr.-Ing. Alfred Mertins

Zweitgutachter: PD Dr. Volker Hohmann

Tag der Disputation: 3.7.2009

Abstract

In this dissertation, methods for an efficient coding of natural sounds are proposed that are based on the concept of "sparse coding" which has been identified as one major mechanism involved in the neurosensory information processing. In the first chapter it is demonstrated that in the MPEG-2/4 AAC audio coding scheme the modified discrete cosine transform (MDCT) can be replaced with a shift-invariant sparse signal model realized by a matching pursuit algorithm. Thereby an improved perceived audio quality was possible, especially at low bitrates. The second chapter addresses the non-trivial problem to select the optimal degree of overcompleteness of a sparse signal model. By using a frame theoretic approach, it is shown that the number $M \geq 100$ of gammatone filters (2.4 filters per ERB) leads to a near-perfect reconstruction of the signal space of natural sounds. In the last two chapters it is demonstrated how a set of significance trees can be used to achieve an effective encoding of sparse coefficients. Using a data-dependent set of significance trees, the proposed coding scheme outperforms the state-of-the-art audio coding scheme MPEG-2/4 AAC for bitrates less than 32 kbps while additionally offering the property of fine-grain bitrate scalability.

Zusammenfassung

In dieser Doktorarbeit werden verschiedene Methoden für eine effiziente Kodierung natürlicher Töne vorgeschlagen, die auf dem Konzept der "spärlichen Kodierung" basieren, welches als ein Hauptmechanismus in der neurosensorischen Informationsverarbeitung identifiziert wurde. In dem ersten Kapitel wird gezeigt, dass in dem Audiokodierungsverfahren MPEG-2/4 AAC die modifizierte diskrete Kosinustransformation (MDCT) durch ein verschiebungsinvariantes spärliches Signalmodel, implementiert durch einen Matching Pursuit Algorithmus, ersetzt werden kann. Dadurch konnte eine verbesserte Audioqualität insbesondere bei niedrigen Bitraten erreicht werden. Das zweiten Kapitel behandelt das nichttriviale Problem der Auswahl der optimalen Übervollständigkeit eines spärlichen Signalmodels. Unter Verwendung der Frame-Theorie wird gezeigt, dass die Anzahl an $M \geq 100$ Gammatonfiltern (2.4 Filter pro ERB) eine fast perfekte Rekonstruktion des Signalraums der natürlichen Töne ermöglicht. In den letzten zwei Kapiteln wird gezeigt wie Signifikanzbäumen genutzt werden können um eine effektive Kodierung spärlicher Koeffizienten zu erreichen. Unter Verwendung einer datenabhängigen Menge von Signifikanzbäumen übertrifft dieses Kodierungskonzept den hochmodernen Audiokodierer MPEG-2/4 für Bitraten unterhalb von 32 kbps während es zusätzlich eine feinabgestimmte Bitratenskalierbarkeit ermöglicht.

Contents

1 Introduction

This dissertation is concerned with an efficient coding of the signal class of natural sounds such as speech or environmental sounds. The working hypothesis is that neurosensory systems are performing a highly optimized signal analysis [8, 7, 75], in particular that the auditory system is realizing a signal model that is specialized in analyzing natural sounds.

Signal models are important in the context of analysis, estimation, compression and synthesis of signals. The earliest theoretical signal analysis model, proposed by Fourier [40], analyzes the frequency content of a signal using the expansion of functions into a weighted sum of sinusoids. Gabor [42] extended this signal model by using shifted and modulated time-frequency atoms which analyze the signal in the frequency as well as in the time dimension. The wavelet signal model, a further improvement presented by Morlet *et al.* [97], uses time-frequency atoms that are scaled dependent on their center frequency. This yields an analysis of the time-frequency plane with a non-uniform tiling. However, the time-frequency atoms used in these signal models normally do not assume an underlying signal structure. As the performance of subsequent processing algorithms depends strongly on how well the fundamental features of a signal are captured, it is favorable to use time-frequency atoms that are specialized to the applied signal class. A data-dependent basis for the signal class of natural sounds can be derived using the independent component analysis (ICA). Thereby derived basis vectors

typically consist of well localized time-frequency atoms [1, 76]. Similar to auditory filters realized by a cochlea [94], the bandwidth of these derived time-frequency atoms increased gradually with the center frequency. This is in contrast to the dyadic increase common for wavelets. It allows however that the temporal asymmetric auditory filters do realize an approximately tight frame within the frequency range of natural sounds (Chapter 3). It is to note that most of the ICA-derived time-frequency atoms did not show the asymmetry in the time domain known to be present in cochlear filter shapes measured experimentally in the auditory nerve [15].

In recent years, studies that applied experimental, computational and theoretical methods [100] could show that one major mechanism involved in the neurosensory information processing is "sparse coding". The sensory information is processed by a large population of neurons of which, especially in the upper parts of the sensory pathways, only a relatively small number of neurons are simultaneously active. In a mathematical context, the sparseness of a vector or matrix is measured by the $L0$ norm, which counts the number of non-zero elements. Hence, optimizing the sparseness of a signal model is achieved by reducing the number of non-zero coefficients representing the signal. This concept of a sparse signal model has gained interest in the signal processing community during the last years [89, 28, 17, 45, 48, 54, 32, 3], as it shows improved performance in signal compression, analysis and denoising tasks [98, 16, 49, 31]. Most sparse signal models assume an additive signal model of the form

$$\mathbf{x}[n] = \sum_{i=1}^{K} \alpha_i \mathbf{d}_i[n] \tag{1.1}$$

with the signal $\mathbf{x} \in \mathbb{R}^{N\times 1}$, the coefficients $\boldsymbol{\alpha} = [\alpha_1 \alpha_2 \dots \alpha_K] \in \mathbb{C}^K$ and the dictionary atoms $\mathbf{D} = [\mathbf{d}_1 \mathbf{d}_2 \dots \mathbf{d}_M] \in \mathbb{C}^{N\times M}$. By postulating the condition $M > N$ and

introducing shift invariance by constructing $\mathbf{D}$ using template atoms and adding all their possible shifts to the dictionary, the encoding of a signal is not unique anymore. This overcompleteness in the time and in the frequency domain allows a sparse signal representation algorithm to search for the sparsest encoding in the infinite number of solutions. Another way to increase the sparseness of a signal model is to increase the correlation of the expansion functions $\mathbf{D}$ with the signal $\mathbf{x}$. Similar to the data-dependent ICA basis, a data-dependent dictionary $\mathbf{D}$ can be learnt that minimizes the $L0$ norm of the coefficients $\boldsymbol{\alpha}$ for a given training set. It has been shown that for natural sounds gammatone time-frequency atoms realize a nearly optimal sparse shift-invariant signal model [117]. In this context it is to note that an overcomplete signal analysis can also be found in the peripheral auditory system. In the frequency domain, an overcomplete signal analysis is realized by the large amount of inner hair cells, for example 3500 inner hair cells in the human cochlea [124]. In the time domain, the activation of spikes in the auditory nerve due to the occurrence of signal energy at a specific frequency and thus a specific place along the basilar membrane [47] is solely threshold triggered and not externally clocked like in block-based signal models. The cochlea performs thereby a shift-invariant and also in the time domain overcomplete signal analysis. This motivates the investigation of a shift-invariant sparse signal model throughout this dissertation and especially the usage of gammatone atoms as expansion functions, as they are also an established model for the signal analysis performed by the human cochlea [104, 105, 15, 22, 23, 103, 19].

The first part (Chapter 2) of this dissertation investigates the feasiblity to replace the block-based signal analysis of a modern audio coding scheme (MPEG-2/4 AAC) with a shift-invariant sparse signal model. In general, the first stage of perceptual audio coding schemes performs a time-frequency analysis of the audio signal. In parallel,

a psychoacoustic model is computing a signal-to-mask ratio from the audio signal, predicting the maximum amount of distortion at each point in the time-frequency plane that is still inaudible. In the next stage, the signal coefficients are quantized, exploiting these perceptual irrelevancies until they meet the desired bitrate after lossless compression [102]. The novelty of this dissertation is to replace the lapped modified discrete cosine transform (MDCT) in the MPEG-2/4 AAC coding scheme with an overcomplete time-invariant sparse signal model that is optimized for natural sounds, while keeping the psychoacoustic model and the quantization algorithm (rate/distortion loop) unchanged. To realize the time-invariant sparse signal model, the time-frequency algorithm matching pursuit [89] was chosen because of its low complexity. Deriving the sparsest representation of a signal has been proven to be NP-hard [28, ch. 2] and is therefore, in general, computationally intractable. It has been shown that the complexity of the matching pursuit algorithm can be reduced for dictionaries that exhibit a special structure [44, 48, 72], i.e. are well localized in time and in frequency. Gammatone functions exhibit such a special structure and a fast matching pursuit algorithm for gammatone functions is presented in Chapter 2 of this thesis.

Matching pursuit is a greedy iterative search algorithm that at the i-th iteration selects the atom having the largest inner product, hence correlation, with the residual $\mathbf{r}_i$:

$$\alpha_{m_i} = \underset{\mathbf{d}_{m_i} \in \mathbf{D}}{\operatorname{argmax}} |\langle \mathbf{r}_i, \mathbf{d}_{m_i} \rangle|^2 \tag{1.2}$$

with m_i being the dictionary index of the selected atom at the i-th iteration. The residual $\mathbf{r}_i$ is defined as the difference between the already encoded signal part and the original signal. The new residual is then computed by removing the selected atom

from the residual:

$$\mathbf{r}_{i+1} = \mathbf{r}_i - \alpha_{m_i}\mathbf{d}_{m_i} \tag{1.3}$$

This is repeated until the signal residuum is small enough or a maximal number of iterations has been reached. An increased correlation of the dictionary atoms $\mathbf{d}_{m_i}$ with the signal results in a faster decline of the residual $\mathbf{r}_{i+1}$. Thus, less iterations (i.e. non-zero coefficients) are needed to encode the signal, which results in an increased sparseness of the signal model. Therefore in this chapter also optimal gammatone parameters achieving the sparsest signal model for English speech are derived.

Matching pursuit is guaranteed to decrease the power of the residual on each iteration only if the atoms of the dictionary span the signal space [89]. However, the non-trivial problem to select the optimal number of gammatone atoms in an overcomplete signal model has not been addressed in any study so far and has been investigated in the second part of this thesis (Chapter 3) using a frame theoretic approach. To achieve an efficient encoding of natural sounds, i.e. to reduce bitcoding and computational costs, it is important to know the smallest number of gammatone atoms needed to achieve a near-perfect reconstruction for the signal class of natural sounds. To further reduce encoding and subband processing costs, it is often favorable to remove the redundancy in an overcomplete signal model by downsampling its subband coefficients. Therefore, another topic addressed in the second part of this dissertation is the concept of multi-rate coding. A bifrequency analysis of the overcomplete gammatone signal model is performed and it is shown how decimation factors can be derived that introduce minimal aliasing distortions.

Even when using the minimal necessary overcompleteness, a shift-invariant sparse signal model still results in a large quantity of subband coefficients for every filter, which

impedes an efficient coding of sounds in terms of memory usage. In the third part of this thesis (Chapter 4 and 5), two compression schemes for sparse data based on bitplane coding and significance trees are proposed. By describing significant coefficients of a bitplane via their position and value information instead of transmitting all values one by one, a high compression performance can be achieved especially for sparse data sets. Encouraged by the success in image compression and the fact that quantized MDCT coefficients are sparse, significance tree related coding techniques have been proposed for audio compression as well [34, 107, 106, 108, 85, 86]. In all of these approaches, the tree structures have been fixed independent of the signal class to be encoded and are based on the assumption that low-frequency components contain more energy than high-frequency components. This assumption, however, does not hold for all frames of real-world audio signals. Thus, fixed trees can only be suboptimal. In Chapter 4, an adaptive tree-based significance mapping technique, the combined significance-tree quantization (CSTQ) [127, 120] is presented. It uses a fixed set of significance trees from which the optimal tree for each frame is selected. An extension of the CSTQ algorithm is developed in Chapter 5 which uses data-dependent significance trees. A scalable compression scheme is proposed that selects for every frame the optimal tree from a dynamically optimized set of data-dependent significance trees. This so-called dynamic significance tree quantization (DSTQ) is based on the concept of data-dependency observed in neurosensory systems [6].

In the context of this dissertation the following peer-reviewed publications have been published:

1. Huan Zhou, Alfred Mertins and Stefan Strahl "An Efficient, Fine-Grain Scalable Audio Compression Scheme" in *Proceedings of the 118th Convention of the Audio Engineering Society*, Barcelona, Spain, Paper No. 6435, May 2005.

2. Stefan Strahl, Huan Zhou and Alfred Mertins "An Adaptive Tree-Based Progressive Audio Compression Scheme" in *Proceedings of the IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA05)*, New Paltz, NY, USA, pp. 219-222, October 2005.

3. Stefan Strahl and Alfred Mertins "Sparse gammatone signal model optimized for English speech does not match the human auditory filters" in *Brain Research*, vol. 1220, pp. 224-233, July 2008.

4. Stefan Strahl and Alfred Mertins "Analysis and Design of Gammatone Signal Models" in *The Journal of the Acoustical Society of America*, vol. 126, no. 5, pp. 2379-2389, November 2009.

5. Stefan Strahl, Heiko Hansen and Alfred Mertins "A Dynamic Fine-Grain Scalable Compression Scheme with Application to Progressive Audio Coding" in *IEEE Transaction on Audio, Speech, and Language Processing*, in print (2010).

2 Sparse gammatone signal model optimized for English speech does not match the human auditory filters[1]

2.1 Abstract

Evidence that neurosensory systems use sparse signal representations as well as improved performance of signal processing algorithms using sparse signal models raised interest in sparse signal coding in the last years. For natural audio signals like speech and environmental sounds, gammatone atoms have been derived as expansion functions that generate a nearly optimal sparse signal model (Smith, E., Lewicki, M., 2006. Efficient auditory coding. Nature 439, 978-982). Furthermore gammatone functions are established models for the human auditory filters. Thus far, a practical application of a sparse gammatone signal model has been prevented by the fact that deriving the sparsest representation is, in general, computationally intractable. In this paper we applied an accelerated version of the matching pursuit algorithm for gammatone dictionaries allowing real-time and large data set applications. We show that a sparse signal model in general has advantages in audio coding and that a sparse gammatone signal model encodes speech more efficiently in terms of sparseness than a sparse modified discrete cosine transform (MDCT) signal model. We also show that the optimal gammatone parameters derived for English speech do not match the human

[1]This chapter has been published in the present form in Brain Research, vol. 1220, pp. 224-233 (2008).

auditory filters, suggesting for signal processing applications to derive the parameters individually for each applied signal class instead of using psychometrically derived parameters. For brain research it means that care should be taken with directly transferring findings of optimality for technical to biological systems.

2.2 Introduction

There is evidence [99, 7, 100] that neurosensory systems encode stimuli by activating only a small number of neurons out of a large population at the same time. This concept of a 'sparse' signal representation has gained interest in the signal processing community in the last years [89, 28, 17, 45, 48, 54, 32, 3] as it shows improved performance in signal compression, analysis and denoising tasks [98, 16, 49, 31]. A sparse signal model indicates the fundamental features of the signal as it necessarily involves expansion functions that are highly correlated with the signal. For natural audio signals like speech and environmental sounds, gammatone atoms have been derived as expansion functions that generate a nearly optimal sparse signal model [76, 117]. Gammatone functions are also known as filters modeling the human cochlea [104, 105] and gammatone filterbanks are applied successfully in simulating the human auditory processing [22, 23, 103, 18]. Deriving the sparsest representation of a signal has been proven to be NP-hard [28, ch. 2] and is therefore, in general, computationally intractable. In this paper we apply an accelerated sparse signal model for gammatone functions which is a specialisation of Matching Pursuit [89]. This time-frequency algorithm computes a sparse signal model from a given dictionary of atoms. At every iteration the dictionary atom that best matches the signal is chosen and removed from the signal. This is repeated until the signal residuum is small enough or a maximal number of iterations has been reached. It

has been shown [44, 48, 72] that the complexity of such an algorithm can be reduced for dictionaries that exhibit a special structure and we apply these results to gammatone dictionaries resulting in a computational complexity of $\mathcal{O}(N \log N)$ per iteration.
The achieved acceleration makes it possible to apply this physiologically motivated signal model in real-time applications like speech coding and analyse its performance in state-of-the-art audio compression schemes like MPEG-4 AAC [63].
The possibility to evaluate the sparse gammatone signal model on a large data set enables the statistical analysis of the selected gammatone parameters for a given sound corpus. According to Barlow's efficient-coding hypothesis [8], the human auditory filters have been optimized under a strong evolutionary pressure to optimally encode the relevant acoustic stimuli. We analyze the TIMIT speech corpus [43] and compare the derived gammatone parameters with the known parameters from psychoacoustic experiments.

2.3 Results

2.3.1 Audio coding

In audio coding schemes such as MPEG-2/4 AAC, the modified discrete cosine transform (MDCT) is used to convert overlapping blocks of the time signal into a frequency-domain representation. With a time shift of N samples, this transform maps $2N$ real numbers onto N real coefficients by using modulated versions of a symmetric window like shown in Figure 2.1a. In the older MPEG-1-layer-3 (MP3) standard, a bank of bandpass filters is used in combination with the MDCT. The symmetry property of the used window results from the Princen-Bradley condition the MDCT has to satisfy

in order to yield a perfect reconstruction transform [90]. In the MPEG-4 AAC coder a sine-shaped and a Kaiser-Bessel derived (KBD) window [101] can be chosen. In both MP3 and AAC the window length can be switched between a short and a long window, which allows the encoder to find the best compromise between a high coding gain in stationary sections (long window) and reduced pre-echoes when the signal contains strong transient components (short window).

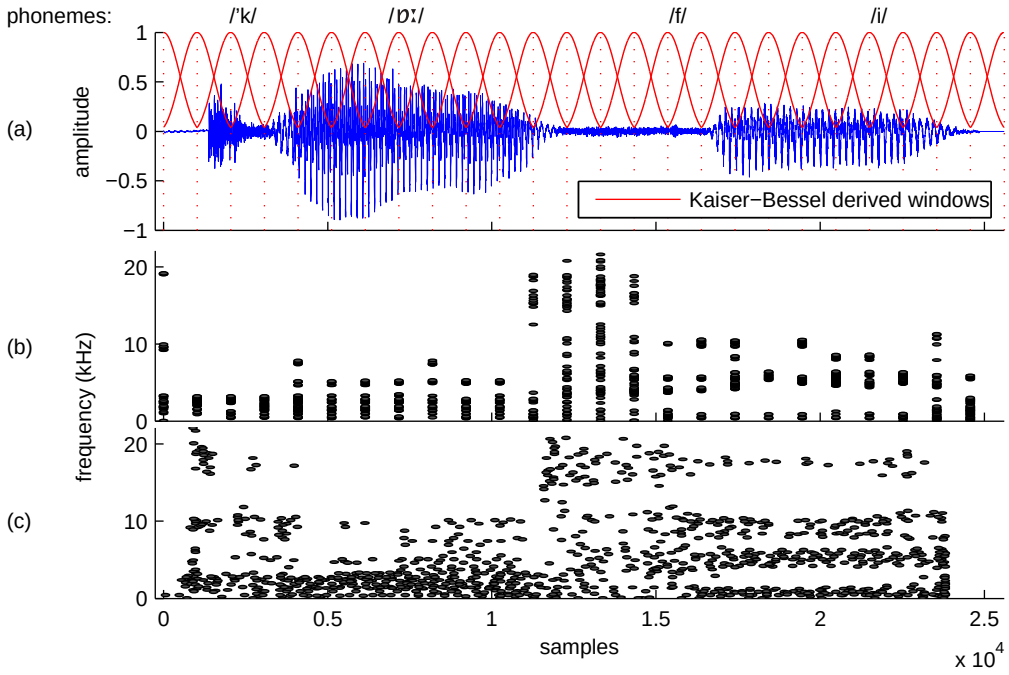


Figure 2.1: (a) Phoneme labeled example from Suzanne Vega's "Tom's Diner" where she sings "coffee". The segments of the overlapping MDCT blocks and the used KBD windows are shown in red. (b) Plot of the 1191 MDCT filterbank coefficients that are not quantized to zero at 16 kbps (c) Plot of the 1018 MDCT matching pursuit coefficients used at 16 kbps

As argued by [115], transforms using a block-wise analysis are very sensitive to small time shifts of the incoming signal and do not encode well transients and periodic components that are located in the middle of the overlap region of two adjacent blocks or window positions.

In this context, it should be noted that the signal conversion into the frequency domain done by the human cochlea is not rigid in time. The occurrence of signal energy at a specific frequency, due to the frequency-to-place mapping along the basilar membrane, results in a deflection of the corresponding inner hair cells, thereby triggering spikes sent over the corresponding auditory nerves to the brainstem. This activation is solely threshold triggered and not externally clocked like in a DCT or MDCT filterbank.

This property of the human auditory system motivates the application of a shift-invariant signal model like matching pursuit [89] allowing arbitrary time positions. It assumes an additive signal model of the form

$$\mathbf{x}[n] = \sum_{i=1}^{K} \alpha_i \mathbf{d}_i[n] \tag{2.1}$$

with the signal $\mathbf{x} \in \mathbb{R}^{N\times 1}$, the dictionary coefficients $\alpha_i \in \mathbb{C}$ and the dictionary atoms $\mathbf{D} = [\mathbf{d}_1 \mathbf{d}_2 \dots \mathbf{d}_M] \in \mathbb{C}^{N\times M}$. The shift-invariance is achieved by constructing $\mathbf{D}$ using templates like gammatone atoms and adding all their possible shifts to the dictionary. Matching Pursuit is a greedy algorithm that first chooses the atom that best approximates the signal. The contribution of this atom is then subtracted from the signal and the process is iterated on the residual. So the task at the i-th iteration is to minimize the residual

$$\mathbf{r}_{i+1}[n] = \mathbf{r}_i[n] - \alpha_i \mathbf{d}_{k_i}[n] \tag{2.2}$$

with $\mathbf{d}_{k_i}[n] \in \mathbf{D}$, k_i being the dictionary index of the atom chosen at the i-th iteration and α_i being the weight describing the contribution of the atom to the signal.

This signal coding paradigm also achieves a sparse signal representation as the

increased time resolution results in an overcomplete representation of the signal space and the encoding of a signal is thereby not unique anymore. This overcompleteness allows the matching pursuit algorithm to search for the sparsest encoding in the infinite number of solutions. In contrast, the MDCT atoms form a basis for the signal space where only one unique representation for a signal exists.

In an initial audio coding experiment, we compared the performance of the matching pursuit approach with the traditional filterbank design using the masking model, scalefactor bands and adaptive quantization of the MPEG-4 AAC audio coding reference implementation. We selected the `castanets.wav` audio signal from the EBU-SQAM audio database [36] due to its transient properties, the TIMIT speech corpus representing the sound class of English speech and the often evaluated music test signal in audio coding, Suzanne Vega's "Tom's Diner" (`svega.wav`). We compared the coding quality of the MDCT filterbank (FB-MDCT) with the matching pursuit signal models using a MDCT (MP-MDCT) and a gammatone dictionary (MP-GAMMA). The results were evaluated using the objective difference grade (ODG) scale [66] computed with an objective prediction method of the perceived audio quality called PEMO-Q [55] (for details see 'Experimental Procedure'). In Figure 2.2 the number of used coefficients per second, the signal-to-noise ratio (SNR) and the ODG of the encoded signals at different bitrates are shown. The matching pursuit algorithm encodes a signal until a given threshold is reached, which was set in this experiment to a fixed SNR for all bitrates (see Table 2.2). The MDCT filterbank in contrast always results in a perfect encoding if no further quantization is applied. In a next step, the matching pursuit respectively filterbank coefficients are encoded with a given bitrate using the masking model, scalefactor bands and adaptive quantization of the MPEG-4 AAC audio coding standard. Thereby the number of used coefficients per second is decreased whenever

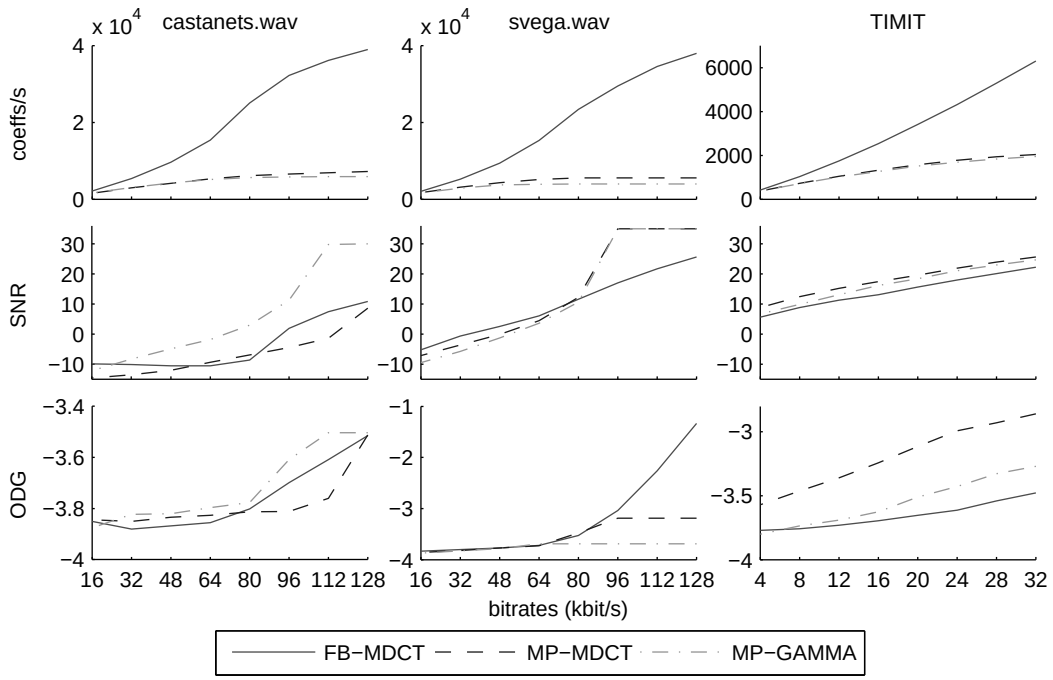


Figure 2.2: The upper row shows the average number of used coefficients per second, the middle row the average signal-to-noise ratio and the lower row the objective difference grade of the encoded signal at different bitrates

a coefficient is quantized to zero. This process can be understood by looking at the coefficients per second and SNR results shown in Figure 2.2 for the Suzanne Vega song. For the highest bitrate, all coefficients of the matching pursuit signal models can be encoded in the given bit budget due to the initial sparse encoding. This results in a signal representation achieving the preset SNR. At reduced bitrates, quantization is needed to achieve the selected bitrate, first reducing the accuracy of the coefficients and later also reducing the number of used coefficients by quantizing small coefficient values to zero. The non-sparse coefficients of the filterbank signal model in contrast need to be quantized for all bitrates. For the castanets test signal, only the MP-GAMMA signal model results in a sparse representation where all coefficients fit into the bit budget at higher bitrates and achieve the preset SNR. The symmetric MDCT atoms cause for

the very transient castanets signal pre-echo artifacts which reduces the SNR. For the TIMIT corpus lower bitrates common for speech coding applications have been chosen, always resulting in a quantisation of the coefficients and an SNR below the preset value. The sparsest encoding of the signal is always achieved by the MP-GAMMA based audio encodings, followed by the MP-MDCT audio encoding and the MDCT filterbank based approach.

For the `castanets.wav` signal the MP-GAMMA audio coder achieves the highest SNR except for the lowest bitrate. This is also reflected in the audio quality. The MP-MDCT signal model encodes in general less signal energy than the FB-MDCT signal model resulting in a lower SNR. This does not directly show in the predicted audio quality, as for low bitrates the MP-MDCT dictionary achieves a better audio quality despite the lower SNR compared with the filterbank based audio coder. Analyzing the audio coding at the lowest bitrates shows that for the gammatone dictionary 86% of the available bit budget is used to encode the position of the coefficients using the standard entropy encoder paradigm, resulting in a much stronger quantization of the coefficient amplitudes compared to the filterbank approach.

For the `svega.wav` signal the SNR of the matching pursuit audio encoding is higher than the filterbank approach for high and moderate bandwidths and slightly lower for low bitrates. The perceived quality of the encoded audio signal is in contrast for high bitrates much better for the FB-MDCT audio coder and for moderate and low bitrates the ODG is almost identical between the three variations of the audio coder. The Suzanne Vega song also includes a significant amount of 'silent' frames having very low signal energy which are not encoded by the matching pursuit signal model due to the sparseness constraint. Analyzing the framewise ODG of the encoded signals shows that the difference in the ODG values between the matching pursuit and filterbank signal

model is due to these frames, which can also be seen in Figure 2.3 showing the SNR and ODG for the example in Figure 2.1. Here the FB-MDCT dictionary achieves the lowest

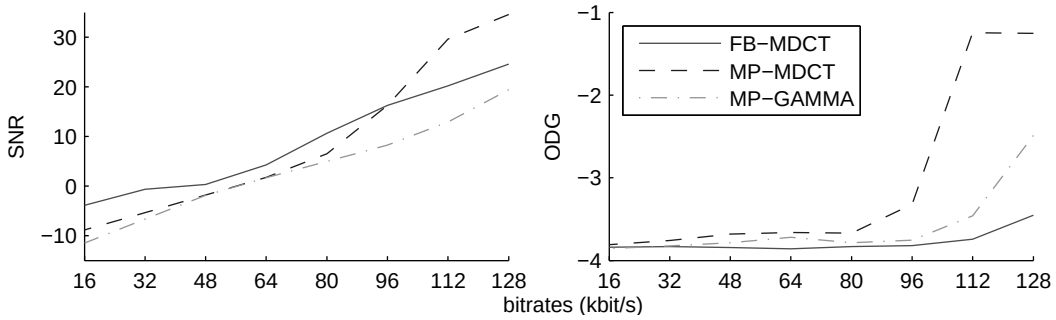


Figure 2.3: The average SNR and ODG for the example in Figure 2.1

perceived audio quality while achieving the highest SNR for low and moderate bitrates. This example shows that while the SNR is a valid measure for general signal coding problems, it is not as significant in audio coding applications as it does not account for psychoacoustic masking effects and does not measure the perceptual distortion. This can be understood by looking at the example in Figure 2.1. The background noise at the end of the example is not encoded in the MP-MDCT signal model reducing the overall SNR of the encoding. The last phoneme `/i/` in contrast is represented using the matching pursuit based audio encoder also in the higher frequency bands above 15kHz where the filterbank approach is not encoding any signal energy for this low bitrate, resulting in a perceptual degeneration of the audio signal. A sparse encoding of a signal results naturally in coefficients with higher coefficient values, which are then not quantized to zero compared to a filterbank approach. So the matching pursuit audio encoder is not only more accurate in time but also generally encodes more high-frequent features than the filterbank approach for a given bitrate.

For the TIMIT speech corpus the SNR of the MP-GAMMA signal model is slightly lower than for the MP-MDCT for high bitrates and drawing near the SNR of the

filterbank implementation for the lower bitrates. The FB-MDCT signal model always achieves the worst SNR. The best perceptual signal quality is always achieved by the MDCT matching pursuit signal model, followed by the MP-GAMMA signal model and the filterbank approach.

The poor performance of the gammatone based audio coder is an unexpected result as gammatone windows have been shown to be optimized to encode speech signals [76, 117]. Despite the fact that the gammatone signal model was always using the lowest number of coefficients to encode a signal to a given SNR, the robustness against quantization errors introduced by the MPEG4 audio encoding scheme showed to be lower compared to the MDCT atoms.

To analyze if using the human parameter values of the gammatone window are optimal for speech we compared the achieved sparseness on variations of the gammatone window using the fast gammatone matching pursuit signal model. The encoding was stopped when an SNR of 20 dB was reached. We did a rigid scan on the parameter space of the gammatone function using the filter orders $\nu = 2, 4, 6, \ldots, 38, 40, 42$ and the damping factors $\lambda = 10, 20, 30, \ldots, 980, 990, 1000$, resulting in 2100 different encodings of the TIMIT database. For the matching pursuit gammatone signal model the minimal

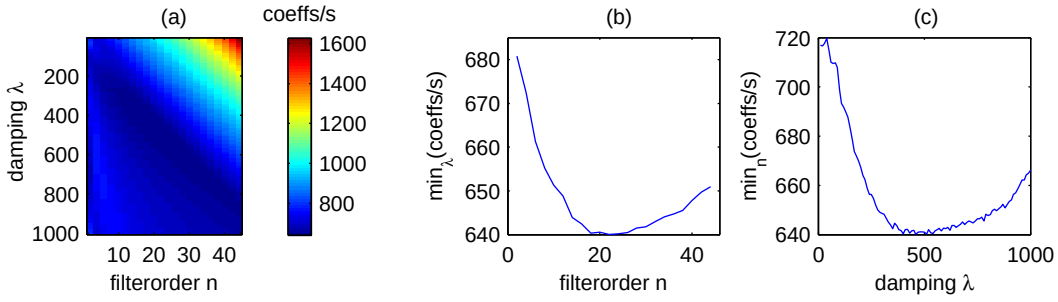


Figure 2.4: (a) average number of coefficients needed per second for a SNR of 20dB for the TIMIT speech corpus (b) minimal number of coeffs/s for a given filter order (c) minimal number of coeffs/s for a given bandwidths

number of coefficients needed to encode an SNR of 20 dB for English speech is, as shown in Figure 2.4, at the filter order $n = 22$ and damping $\lambda = 460$, resulting in 640.1 coeffs/s. We retested the audio coder with these optimized values. As shown in

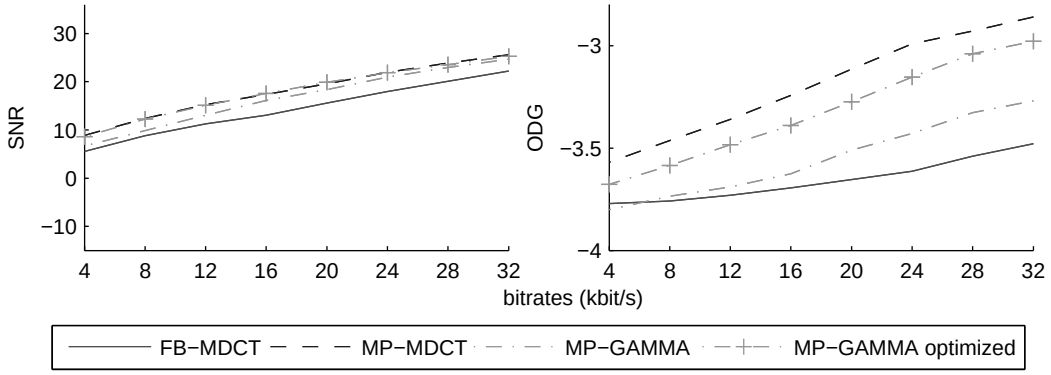


Figure 2.5: The left column shows the SNR, the right the ODG for the TIMIT sound corpus using the different encoding schemes at different bitrates

Figure 2.5, the optimized gammatone dictionary achieves now an SNR for the audio encoded TIMIT speech corpus which is almost identical to the MP-MDCT audio codec. This is not reflected in the perceived audio quality, where the MP-GAMMA albeit its sparser encoding is showing a higher impact of quantisation errors on the perceived audio quality than the MDCT dictionary. Informal listening tests showed that the gammatone dictionary suffered from stronger musical tones artifacts compared to the MDCT dictionary. It should be kept in mind that this is an initial audio coding experiment. For example there is a trade-off between the number of initial coefficients generated by the matching pursuit signal model and the consequently needed quantization of these coefficients to achieve the given bitrate. This has not been explored here, the stopping condition of the iterative matching pursuit algorithm was preset to a fixed SNR. More psychoacoustically motivated stopping rules could result in a better audio encoding quality. Additionally the lossless compression stage has

been implemented using the standard entropy encoder paradigm to be able to directly compare the different signal models within the MPEG4-AAC audio coding scheme. We have shown that using a significance-tree coder [120] brings advantages for sparse data and shows good performance for audio coding. Furthermore the quantization algorithm can be optimized for a matching pursuit signal model [46, 41].

To further investigate why the gammatone matching pursuit signal model results in a sparser encoding of the TIMIT database than the MDCT matching pursuit signal model for a given SNR, we adapted the gamma-window parameters slowly from an asymmetric to an approximately symmetric window while keeping the maximum of the window fixed (see Table 2.1). We analyzed its performance on the TIMIT speech database encoding up to an SNR of 20dB. The number of coefficients per second

signal model	n	λ	skewness	coeffs/s
MP-MDCT			0.0	693.2
MP-GAMMA	2	31.25	-1.74	799.0
	4	93.9	-0.71	727.5
	6	156.25	-0.09	682.0
	8	218.9	0.17	663.5
	10	281	0.35	653.0
	12	344	0.49	648.9
	14	406	0.61	651.2
	16	469	0.74	651.5
	18	531	0.82	655.5

Table 2.1: Coefficients per second needed to achieve an SNR of 20dB for gammatone windows with different skewness and the TIMIT sound corpus

decreased from the approximate symmetric gammatone window with a skewness of -0.09 to a minimum at a skewness of 0.49. This is consistent with the earlier derived optimal gammatone parameters $n = 22, \lambda = 460$ which result into a skewness of 0.45. This indicates that a positive skew resulting in an asymmetry is one of the important

properties of the gammatone dictionary that leads to an increased sparseness for speech compared to the symmetric Kaiser-Bessel derived window.

2.3.2 Physiological signal model

We conducted a further experiment using a very large dictionary of gammatone atoms with center frequencies ranging from 15.625 Hz to 8000 Hz, increased in 15.625 Hz steps, and damping parameters $\lambda = 2\pi bERB(f_c)$ ranging from 100 to 5950, increased in steps of 50, resulting in a dictionary size of 966,656,000 atoms per second. Figure 2.6a shows

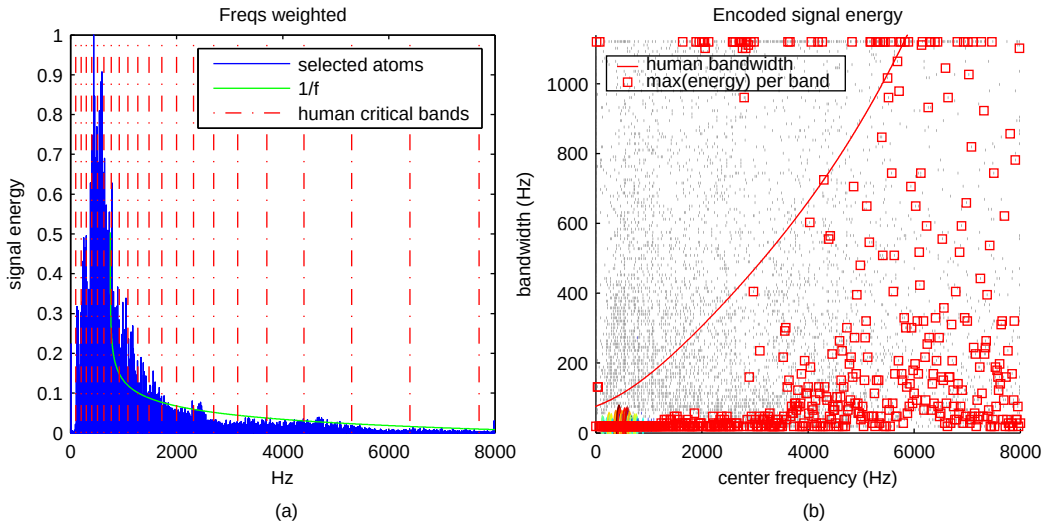


Figure 2.6: (a) Frequency distribution of selected atoms (b) Encoded signal energy per bandwidth and frequency of selected atoms to encode 20dB SNR of the TIMIT speech corpus using $\lambda \in \{100, 150, \cdots, 5900, 5950\}$

the center-frequency distribution of the selected atoms, which follows the $1/f$ law normally found for natural signals [9]. The selected gammatone bandwidth parameter b was mostly chosen as 0.19 which differs from the human value of 1.019 [58]. The selected bandwidths for every frequency band are shown in Figure 2.6b, where the size of the datapoint and its color represents the amount of signal energy that is encoded

using this parameters. Atoms encoding less than one percent but more than one tenth of a percent of the TIMIT sound corpus are plotted in gray. The bandwidth parameter used to encode the most energy in the according frequency band is marked by a red square. The matching pursuit algorithm selected mainly atoms with bandwidths below 100Hz. Also the maximal bandwidth was frequently chosen. It can be noted that the bandwidths encoding the most energy of the signal per frequency band mainly stay below the human bandwidths [128]. This highly overcomplete dictionary encodes the TIMIT database with an average of 543 coeffs/s.

We further tested if an encoding of the English speech database into a sparse representation limited to 21 different bandwidths for all frequencies with the human values

$$\lambda = 193, 262, 331, ..., 3477, 4169, 4998$$

[128] would result in any physiologically known parameter values. Figure 2.7a shows

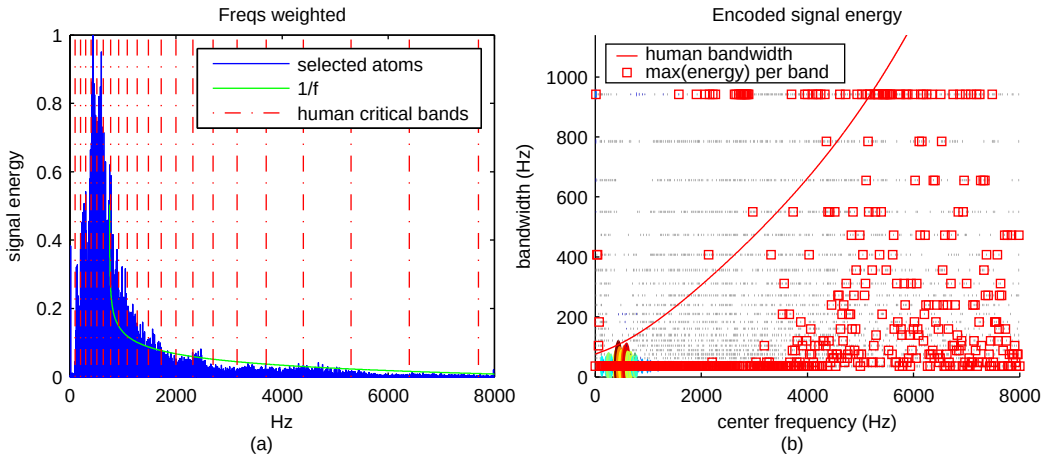


Figure 2.7: (a) Frequency distribution of selected atoms (b) Encoded signal energy per bandwidth and frequency of selected atoms to encode 20dB SNR of the TIMIT speech corpus using human $\lambda \in \{193, 262, \cdots, 4169, 4998\}$

again a frequency distribution following the $1/f$ law and most of the signal energy was

now encoded using $b = 0.342$. Figure 2.7b shows a similar distribution of the selected atoms like in Figure 2.6b. Again, mainly atoms with small bandwidth are preferred. And the most selected bandwidth per frequency is again widening at higher frequencies but staying below the human bandwidth. This fixed bandwidth dictionary encodes the TIMIT database with an average of 607 coeffs/s.

The selection of mainly long dictionary atoms having a small filter bandwidth compared to the human auditory filters is coherent with the signal structure of the TIMIT speech corpus. We computed an average phoneme length of 72.8ms for the TIMIT database and the filter lengths mainly selected by the matching pursuit algorithm are the two longest atoms with 116ms and 77.3ms, as they result in the highest correlation with the signal. The occasional selection of short atoms having a long filter bandwidth can be attributed to short signal parts like consonants and to artifacts generated by the matching pursuit algorithm due to its iterative signal decomposition.

2.4 Discussion

The hypothesis driving the present study is an application of Barlow's efficient coding hypothesis [8] for audio signal coding. The main efficiency measure in biological systems is the number of spikes needed to transmit a representation of the perceived signal [75]. This corresponds in a computer signal model with the number of coefficients used to encode a signal or in other words, how sparse a signal encoding is. One way of increasing the sparseness of an encoding is to increase the correlation of the analyzing filter respectively dictionary atoms with the signal class. We could verify previous results [76, 117] showing that gammatone atoms have an increased correlation with the English speech, achieving a higher sparseness compared to MDCT atoms. One

of the main properties leading to the increased sparseness is the asymmetric time envelope of a gammatone atom as shown in Table 2.1. This can be understood by the fact that most natural sounds are asymmetric in time, exhibiting a short transient followed by an exponential damped oscillation. This also yields benefits regarding the matching pursuit. The algorithm picks the most energetic atom and for a dictionary with symmetric atoms, it will choose an atom that has also support before the actual start of the attack of the sound. Subtracting this atom from the signal will result in a pre-echo artifacts, creating an artificial signal component just before the transient. The asymmetry of the gammatone window prevents such pre-echo artifact.

Furthermore, the envelope asymmetry indicates that the sparseness constraint is more important for the neurosensory system than a signal analysis achieving a perfect time-frequency resolution. A dictionary atom can not be arbitrarily concentrated in both time and frequency. [42] has shown that, given Heisenberg's uncertainty principle, a symmetric modulated Gaussian window achieves optimal joint time-frequency resolution. For the visual system, such two-dimensional Gabor atoms have been derived as expansion function that generate a nearly sparse signal model and have also been verified in the visual cortex [99]. Compared to a Gabor atom, a gammatone atom has an enlarged analysing window area in the time-frequency plane of a factor of $\sqrt{\frac{2n-1}{2n-3}}$, n being the filter order of the gammatone [118]. The gammatone dictionary showing the highest correlation with the TIMIT database and thereby achieving the sparsest encoding has a higher filter order and thus a better joint time-frequency resolution than a signal model using the human physiological parameters. Gammatone functions with non-human parameters for the TIMIT speech corpus have also been derived by [116], optimizing a randomly initialised dictionary using a gradient search algorithm. In this study we applied a full search over the parameter space of the gammatone

function, showing that only one local minimum exists for the TIMIT speech corpus. The non-human parameters can be explained by the fact, that sparseness is not the only constraint that shaped the auditory system. Consequently, a pure matching pursuit model is not a valid correspondence to the human auditory filter, explaining why the parameters achieving the sparsest encoding for the TIMIT sound corpus do not resemble the human physiological data.

Another important effect is the physiological size constraints. Sparseness is achieved by an overcomplete signal model, whose atoms have overlapping analysis windows in the time-frequency domain. To increase the sparseness of a given signal model, the overlap of theses areas needs to be increased. It has been shown that the increase in length of the auditory epithelia during phylogeny is greater than the increase in the upper frequency, especially in birds and mammals [91], but a momentous increase in frequency resolution is impeded by the size constraint of the hair cell and the cochlear length itself. Consequently, the sparse encoding of an auditory stimulus at the stage of the cochlear is achieved mainly by the high time resolution of its shift-invariant signal model. Analogous to the visual system, where the edge detector filters predicted by a sparseness constraint can be found in the primary visual cortex [99], it can be assumed that in the auditory system the sparse coding paradigm will also have an increased influence in the later stages of the auditory pathway compared to the early stages.

Except for [117], all audio coding applications using a gammatone signal model [4, 37, 121] applied human parameters and a block-based filterbank model. In general, it has been shown that audio coding applications using a sparse signal model like matching pursuit can have advantages compared to critical sampling signal models like filterbanks or wavelet analysis [49, 27, 72, 117]. Using the union of a MDCT and modified discrete sine transform (MDST) as a signal model, Davies et al. have shown

that a two-fold oversampling in the frequency domain results for a transient guitar solo test signal in higher SNRs compared to a MDCT signal model [27]. Smith et al. showed for English speech alike an increased SNR compared to a wavelet or Fourier transform using a signal model that is highly overcomplete in time [117]. Replacing the normal quantization with a psychoacoustic masking model, scalefactor bands and adaptive quantization, we measured also an increased SNR and perceived audio quality compared to the block-based signal models for the transient castanets signal and the English speech corpus. The sparseness constraint effects the distribution of the signal energy to few coefficients with high coefficient amplitudes and many coefficients with near zero or zero amplitude. This leads to a distribution of the quantization errors which are mostly either below the absolute hearing threshold or at high sound pressure levels, which is advantageous due to the human logarithmic scale of sound intensity. Additionally fewer coefficients are quantized to zero compared to a filterbank approach, which preserves more of the original signal structure in the quantization process. The synthesis of the atomic signal decomposition introduces also artificial patterns like musical tones, generating a disturbing tonal percept due to equidistant structures on the frequency scale. These artifacts need to be addressed using a postprocessing step in the audio coding design.

2.5 Conclusion

A matching pursuit gammatone signal model for the English speech using the TIMIT database has been analyzed, showing that, compared to MDCT dictionaries, gammatone dictionaries achieve a sparser encoding for the TIMIT database, indicating that the gammatone atoms are expansion functions that are higher correlated with the English

speech class. We also showed that a shift-invariant matching pursuit signal model has advantages in audio-coding applications and that a gammatone matching pursuit signal model results in better perceived audio quality for very transient signals due to their asymmetric filter shape. A full search over the gammatone filter parameter space showed that the human auditory system can not be directly compared to a matching pursuit signal model and that the optimal parameters are not identical to the human physiological values. We showed that the asymmetric filtershape of the cochlear filter can be predicted assuming a sparseness constraint on the signal coding.

2.6 Experimental Procedure

2.6.1 Gammatone signal model

The gammatone signal model describing the human auditory filter response is defined as [105]

$$g_t(t) = at^{n-1}e^{-\lambda t}e^{2\pi i f_c t} = at^{n-1}e^{-2\pi b ERB(f_c)t}e^{2\pi i f_c t} \tag{2.3}$$

with the amplitude a, the filter order n and $\lambda = 2\pi b ERB(f_c)$ being the damping factor where b defines the proportion to the equivalent rectangular bandwidth (ERB) of the auditory filter which is defined for moderate sound pressure levels [96] as $ERB(f_c) = 24.7 + 0.108 * f_c$ for a center frequency f_c. For humans the parameters $n = 4$ and $b = 1.019$ have been derived using notched-noise masking data [58].

2.6.2 Fast Matching Pursuit for gammatone signal model

Every real-valued atom $d_{\omega,\phi}$ with the frequency ω and the phase ϕ can be associated with a complex atom d_ω and its conjugate $\overline{d_\omega}$. It is

$$d_{\omega,\phi} = \frac{K_{\omega,\phi}}{2}\left(e^{i\phi}d_\omega + e^{-i\phi}\overline{d_\omega}\right) \tag{2.4}$$

with $K_{\omega,\phi}$ being a normalization factor. The set of atoms $d_{\omega,\phi}$ where only the phase varies lies in the subspace that is spanned by d_ω and $\overline{d_\omega}$. So the orthogonal projection $P_{\mathbf{V}_\omega} r_i$ of the residuum r_i onto this subspace $\mathbf{V}_\omega := span\{d_\omega, \overline{d_\omega}\}$ results in a vector lying in the direction of the real atom $d_{\omega,\phi}$ having the optimal phase. This variation is called *Molecular Matching Pursuit* [48] as selecting the best real atom $d_{\omega,\phi}$ is equivalent to finding the best *di-atomic molecule* $\mathbf{V}_\omega$ with

$$\sup_{\omega,\phi} |\langle r_i, d_{\omega,\phi}\rangle|^2 = \sup_{\omega}\sup_{\phi} |\langle r_i, d_{\omega,\phi}\rangle|^2 = \sup_{\omega} ||P_{\mathbf{V}_\omega} r_i||^2 \tag{2.5}$$

Using the biorthogonal basis $d_\omega^\otimes, \overline{d_\omega^\otimes}$ of $\mathbf{V}_\omega$ with

$$d_\omega^\otimes = \frac{1}{1 - |\langle \overline{d_\omega}, d_\omega\rangle|^2}\{d_\omega - \langle d_\omega, \overline{d_\omega}\rangle \overline{d_\omega}\} \tag{2.6}$$

the orthogonal projection on a di-atomic molecule is computed by

$$P_{\mathbf{V}_\omega} r_i = \langle r_i, d_\omega\rangle d_\omega^\otimes + \langle r_i, \overline{d_\omega}\rangle \overline{d_\omega^\otimes} \tag{2.7}$$

and it follows

$$||P_{\mathbf{V}_\omega} r_i||^2 = \frac{2Re\left\{|\langle r_i, d_\omega\rangle|^2 - \langle d_\omega, \overline{d_\omega}\rangle \langle r_i, d_\omega\rangle^2\right\}}{1 - |\langle \overline{d_\omega}, d_\omega\rangle|^2} \tag{2.8}$$

The orthogonal projection of the real-valued signal on the space spanned by a complex gammatone atom and its conjugate transpose can be computed completely in the frequency domain using the fast Fourier transformation (FFT), resulting in complexity of $\mathcal{O}(N \log N)$ instead of $\mathcal{O}(N^2)$ per matching pursuit iteration with N being the length of the analysed signal part. The results in this paper have been computed using the free available Matching Pursuit Toolkit [50] which conducts an initial analysis of the signal and only recomputes in the next iteration the changed signal part, resulting in an overall complexity of $\mathcal{O}(L \log L) + K \cdot (2N-1)\mathcal{O}(N \log N)$ with L being the signal length, K the number of iterations and N the atom length.

It is

$$\langle r_i, d_\omega \rangle = \sum_{t=0}^{N-1} r_i[t] t^{n-1} e^{-\lambda t} e^{-2\pi i \frac{\omega t}{N}} dt = \mathcal{FFT}_\omega(r_i[t] t^{n-1} e^{-\lambda t})$$

and

$$\langle d_\omega, \overline{d_\omega} \rangle = \sum_{t=0}^{N-1} \left(t^{n-1} e^{-\lambda t}\right)^2 e^{2\pi i \frac{2\omega t}{N}} dt = \mathcal{FFT}_{-2\omega}(\left(t^{n-1} e^{-\lambda t}\right)^2).$$

In the audio-coding experiment we omitted the phase information of the gammatone signal model for a valid comparison with the MDCT filterbank signal model. For only real-valued atoms we have $\langle d_\omega, \overline{d_\omega} \rangle = 1$ simplifying the projection to

$$\langle r_i, d_\omega \rangle d_\omega = \pm \alpha d_\omega$$

Our gammatone atom implementation will be available on the official MPTK web page.

2.6.3 Audio coding

We used the perceptual model, scalefactor bands and adaptive quantization algorithms from the MPEG4 AAC reference implementation [63, 102]. The final noiseless coding stage has been adapted for the sparse overcomplete matching pursuit signal models by adding a run-length encoding step before the entropy encoder similar to the encoding step in the JPEG standard.

We used the following signals and settings:

audiosignal	resolution	samplingrate	length	SNR threshold
`castanets.wav`	16 Bit	48.0 kHz	7s 939ms	30 dB
`svega.wav`	16 Bit	44.1 kHz	20s 675ms	35 dB
`TIMIT`	16 Bit	16 kHz	5h 35min	30 dB

Table 2.2: Audio coding settings

For the Suzanne Vega music sample `svega.wav` an increased SNR threshold of 35dB was necessary to achieve a sufficient coding quality due to its more complex signal structure.

We predicted the perceived audio quality of the encoded audio signals relative to the uncoded signal using a model of auditory perception (PEMO-Q) [55]. The estimated perceived audio quality is mapped to a single quality indicator, the Objective Difference Grade (ODG) [66]. This is a continuous scale from 0 for "imperceptible impairment", -1 for "perceptible but not annoying impairment", -2 for "slightly annoying impairment", -3 for "annoying impairment" to -4 for "very annoying impairment".

We tested the common bitrates 128, 112, 96, 80, 64, 32, 16 kbps for music and 32, 28, 24, 20, 16, 12, 8, 4 kbps for speech. The matching pursuit signal models were restrained to real-valued atoms to allow a valid comparison to the real-valued MDCT filterbank of the AAC reference implementation. The initial MP-GAMMA signal model used a filter

order of 4 and a damping factor of 1000 corresponding to the human filter bandwidth at 1.2 kHz. The skewness of an atom waveform was computed by $y = \frac{E(x-\mu)^3}{\sigma^3}$.

3 Analysis and Design of Gammatone Signal Models[1]

3.1 Abstract

An established model for the signal analysis performed by the human cochlea is the overcomplete gammatone filterbank. The high correlation of this signal model with human speech and environmental sounds (Smith, E. and Lewicki, M. (**2006**). "Efficient auditory coding", Nature **439**, 978-982), combined with the increased time-frequency resolution of sparse overcomplete signal models, makes the overcomplete gammatone signal model favorable for signal processing applications on natural sounds. In this paper a signal-theoretic analysis of overcomplete gammatone signal models using the theory of frames and performing bifrequency analyses is given. For the number of gammatone filters $M \geq 100$ (2.4 filters per ERB), a near-perfect reconstruction can be achieved for the signal space of natural sounds. For signal processing applications like multi-rate coding, a signal-to-alias ratio can be used to derive decimation factors with minimal aliasing distortions.

3.2 Introduction

The earliest theoretical signal analysis model, proposed by Fourier [40], analyzes the frequency content of a signal using the expansion of functions into a weighted sum

[1]This chapter has been published in the present form in JASA, vol. 126, no. 5, pp. 2379-2389 (2009).

of sinusoids. Gabor [42] extended this signal model using shifted and modulated time-frequency atoms which analyze the signal in the frequency as well as in the time dimension. With the wavelet signal model, a further improvement was presented by Morlet *et al.* [97] using time-frequency atoms that are scaled dependent on their center frequency. This yields an analysis of the time-frequency plane with a non-uniform tiling. The time-frequency atoms used in these signal models normally do not assume an underlying signal structure. As the performance of subsequent processing algorithms depends strongly on how well the fundamental features of a signal are captured, it is favorable to use time-frequency atoms that are specialized to the applied signal class. In this paper we are concerned with the signal class of natural sounds such as speech or environmental sounds, which have been found to be highly correlated with gammatone time-frequency atoms [76, 117]. The signal-dependent properties of gammatone atoms are their non-uniform frequency tiling of the time-frequency plane and their asymmetric envelope [119]. A gammatone filterbank is furthermore an established model for the human auditory filters [104, 105, 22, 23, 103, 19]. Several analysis-synthesis systems have been proposed using gammatone filters in the analysis and time-reversed filters in the synthesis stage [73, 74, 81, 37], including low-delay [53] and level-dependent asymmetric compensation[57] concepts.

Overcompleteness in signal models has advantages in signal coding applications. It enables sparse signal models like matching pursuit [89] to search for the sparsest signal representation from the resulting infinite number of possible encodings. Overcompleteness further introduces a robustness towards noise [21, 11]. Generally, the choice of the number of time-frequency atoms in a signal model, hence the choice of overcompleteness, is nontrivial. In this paper we are therefore also concerned with the trade-off between the achieved performance in the subsequent processing algorithms and the introduced

computational load. To derive the minimal number of time-frequency atoms needed to realize an overcomplete gammatone signal model that can adequately analyze the signal space, we use the theory of frames [33, 26, 24, 25] which is a generalization of signal representations based on transforms and filterbanks. A second parameter that can control the overcompleteness of the gammatone signal model is the number of removed analysis filter coefficients. Such a decimation of the filter coefficients introduces aliasing distortions that should not only be kept to a minimum, but should also be steered to cancel out in the synthesis stage of the filterbank. Therefore we performed a bifrequency analysis [20] in addition to a frame-theoretic analysis of overcomplete decimated gammatone signal models. We show how a signal-to-alias ratio can be used to derive optimal sets of decimation factors with minimal aliasing distortions at a given total decimation factor.

The paper is organized as follows: In the next section we introduce the analyzed overcomplete gammatone signal models. In Section 3.4 we present a frame-theoretic analysis of a non-decimated and a decimated overcomplete gammatone signal model by performing an eigenanalysis of the frame operator [12]. We further show how these results can be used to select the optimal number of atoms for an overcomplete gammatone signal model. In Section 3.5 we show how optimal decimation factors with minimized distortion artifacts can be derived using the bifrequency system analysis [20]. We then analyze these theoretically derived optimal parameters in Section 3.6 in several audio coding examples.

Notation

Matrices and vectors are printed in boldface. $||\cdot||$ means the Euclidian norm of a vector. $\langle\cdot,\cdot\rangle$ is the inner product of a vector space. $\mathbb{Z}$ is the set of all integers, $\mathbb{R}$ is the set of all real, and $\mathbb{C}$ is the set of all complex numbers. $[a,b] := \{x|a \leq x \leq b\}$ represents the set of all numbers between and including a and b. The superscript * denotes the complex conjugate of a complex number and the superscript ^{H} the conjugate transposition of a complex m-by-n matrix. The asterisk $*$ denotes convolution. The argument of the maximum of a function $f(x)$ is denoted as $\underset{x}{\operatorname{argmax}} f(x)$.

3.3 Overcomplete Gammatone Signal Model

3.3.1 Gammatone Function

In 1960, Flanagan used a gammatone function as a model of the basilar membrane displacement in the human ear [39]. Johannesma further showed in 1972 that a gammatone filter can be used to approximate responses recorded from the cochlear nucleus in the cat [70]. In 1975, de Boer used a gammatone function to model impulse responses from auditory nerve fiber recordings in the cat, which have been estimated using a linear reverse-correlation technique [29]. The term "Gamma-tone" was introduced 1980 by Aertsen and Johannesma [2]. Patterson *et al.* stated 1988 that the gammatone filter also delineates psychoacoustically determined auditory filters in humans [105]. A gammatone filter is defined as

$$\gamma[n] = an^{\nu-1}e^{-\lambda n}e^{2\pi i f_c n} \tag{3.1}$$

with the amplitude a and the filter order ν. The damping factor λ is defined as $\lambda = 2\pi b ERB(f_c)$ with the center frequency f_c. The parameter b controls the bandwidth of the filter proportional to the equivalent rectangular bandwidth (ERB) of a human auditory filter. For humans, the parameters $\nu = 4$ and $b = 1.019$ have been derived using notched-noise masking data [58]. For moderate sound pressure levels, Moore *et al.*[96] estimated the size of an ERB in the human auditory system as $ERB(f_c) = 24.7 + 0.108 f_c$. The center frequencies of the gammatone filters are equally spaced on the ERB frequency scale [95]. The scale is defined as the number of ERBs below each frequency with $ERBS(f_c) = 21.4 \log_{10}(0.00437 f_c + 1)$. This non-uniform distribution of the center frequencies (see Figure 3.1) correlates with the $1/f$ distribution of frequency energy found in natural signals [9]. It is one of the signal-dependent features of a gammatone signal model. The frequency-dependent bandwidth resulting in narrower filters at low frequencies and broader filters at high frequencies is also an important feature of the gammatone time-frequency atoms. In Section 3.4 we will show that this enables the signal model to form a snug frame. The third signal-dependent feature of gammatone time-frequency atoms is the asymmetric envelope of the gammatone function [119], which can also be found in natural sounds, exhibiting a short transient followed by an exponentially damped oscillation.

3.3.2 Overcomplete Gammatone Signal Model

To analyze overcomplete gammatone signal models we first have to define a corresponding discrete signal processing system (Figure 3.2). The signal $x[n]$ is analyzed with a filterbank where $h_m[n], m \in [0, M-1]$ denotes the impulse responses of M gammatone filters. This splits the full-band signal $x[n]$ into M frequency bands (subbands). In

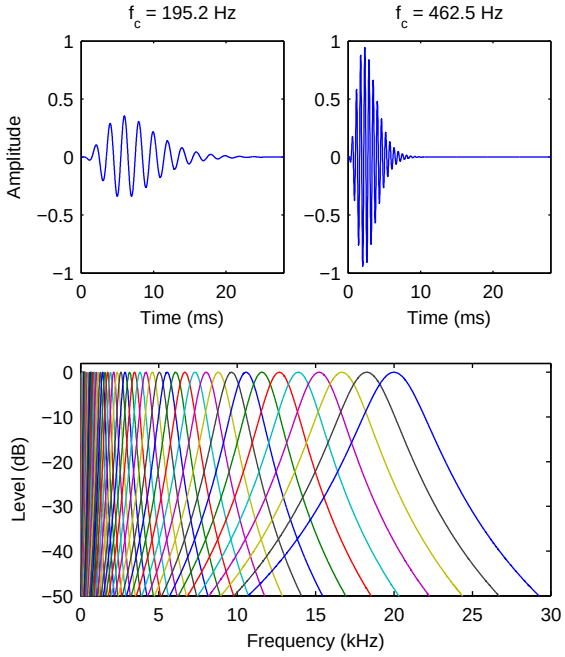


Figure 3.1: (color online) In the upper row the waveforms of two gammatone filters are plotted. The lower row shows the magnitude frequency response of $M = 50$ gammatone filters that are equally distributed along the ERB scale from 20 Hz - 20 kHz.

many signal processing applications these subbands are subsampled by decimation factors N_m to remove redundancy from the internal representation and thereby reducing the overcompleteness of the signal model. For the maximally decimated case with $\frac{1}{N_0} + \cdots + \frac{1}{N_{M-1}} = 1$, a critical sampling is realized, meaning that the amount of data

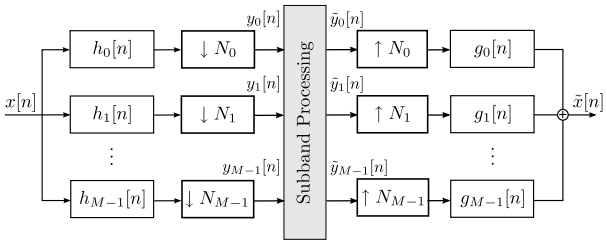


Figure 3.2: Discrete signal processing system used to analyze the overcomplete gammatone signal models.

(samples per second) in the transformed domain and for the original signal are the same. For $\sum_{m=0}^{M-1} \frac{1}{N_m} > 1$ the signal model is overcomplete and there are more subband coefficients $y_m[n]$ per time unit than input samples $x[n]$. All subband coefficients $y_m[n]$ are then routed into a subband processing block. In this block, further operations could be performed, like, for example, a quantization of the subband coefficients controlled by a psychoacoustic model or a sparse signal model algorithm like matching pursuit (see Appendix 3.9.1). After the subband processing, the signal $\tilde{x}[n]$ is reconstructed from the M processed subband signals $\tilde{y}_m[n]$ by upsampling with N_m, followed by the synthesis filterbank with the filters having impulse responses $g_m[n], m \in [0, M-1]$.

The analysis presented in this paper is applicable for two different variations of the gammatone signal model. The first variation uses gammatone analysis filters $h_m = \gamma[n]$ and reversed gammatone synthesis filters $g_m = \gamma[-n]$. This is the most commonly used design, for example in audio coding applications [73, 74, 37]. The second variation uses reversed gammatone analysis filters $h_m = \gamma[-n]$ and gammatone synthesis filters $g_m = \gamma[n]$. This system can be used to perform a fast matching pursuit analysis with a gammatone dictionary (see Appendix 3.9.1). By choosing the synthesis filters as the time-reverse of the analysis filters the overall filterbank response has a linear phase in both designs.

A gammatone signal model is normally designed to cover only a limited frequency range [22, 23, 103, 19, 73, 74, 4, 37]. Consequently, the analyses in this paper have been conducted using such bandlimited gammatone signal models. We distributed the center frequencies of the gammatone filters equally spaced on the ERB scale within the interval $f_c \in [20, 20000]$ Hz, which represents the approximated human hearing range [59].

3.4 Frame-Theoretic Analysis of an Overcomplete Gammatone Signal Model

In this section, we will perform a frame-theoretic analysis of the overcomplete gammatone signal model. We will introduce the theory of frames and use it to evaluate the properties of the corresponding frame of a non-decimated and a decimated gammatone signal model. All calculations have been performed with a sampling rate of 96 kHz, and the length of the impulse responses $h_m[n]$ and $g_m[n]$ was 8192 samples or 85.3ms, respectively.

3.4.1 The Theory of Frames

The theory of frames provides a mathematical framework to analyze overcomplete signal models [26, 24, 25]. A *frame* of a vector space $\mathbf{V}$ is a set of vectors $\{\mathbf{e}_m\}$ which satisfy the following *frame condition*[25]

$$A||\mathbf{v}||^2 \leq \sum_m |\langle \mathbf{v}, \mathbf{e}_m \rangle|^2 \leq B||\mathbf{v}||^2 \; \forall \, \mathbf{v} \in \mathbf{V} \tag{3.2}$$

with the *frame bounds* $A > 0$ and $B < \infty$. Frames can be seen as a generalization of bases, as the set $\{\mathbf{e}_m\}$ is allowed to be linearly dependent and (3.2) implies that the set $\{\mathbf{e}_m\}$ must span the vector space $\mathbf{V}$. Otherwise it would follow $A = 0$ from $\langle \mathbf{v}, \mathbf{e}_m \rangle = 0$ for $\mathbf{v} \in \mathbf{V} \setminus \text{span}\{\mathbf{e}_m\}$.

The frame condition can also be written as $A||\mathbf{v}||^2 \leq \langle \mathbf{S}\mathbf{v}, \mathbf{v} \rangle \leq B||\mathbf{v}||^2$ with $\mathbf{S}$ being the *frame operator* defined as

$$\mathbf{S}\mathbf{v} = \sum_m \langle \mathbf{v}, \mathbf{e}_m \rangle \mathbf{e}_m. \tag{3.3}$$

The frame bound A is the essential infimum and the frame bound B is the essential supremum of the eigenvalues of $\mathbf{S}$ [25]. A frame is called *tight* if $B/A = 1$ and *snug* if $B/A \approx 1$. The advantage of a tight frame is that perfect reconstruction can be done by the frame itself:

$$\mathbf{v} = \frac{1}{A} \sum_{m} \langle \mathbf{v}, \mathbf{e}_m \rangle \mathbf{e}_m \ \forall \, \mathbf{v} \in \mathbf{V} \tag{3.4}$$

The frame bounds for the in this study analyzed discrete signal processing system as shown in Figure 3.2 are given by the following inequality:

$$A||\mathbf{x}||^2 \leq \sum_{m=0}^{M-1} \sum_{k=-\infty}^{\infty} |\langle \mathbf{x}, \mathbf{h}_{m,k} \rangle|^2 \leq B||\mathbf{x}||^2 \ \forall \, \mathbf{x} \in \ell^2(\mathbb{Z}) \tag{3.5}$$

with $m \in [0, M-1], k \in \mathbb{Z}$ and the vectors $\mathbf{h}_{m,k}$ containing the filter coefficients $h_m(kM - n)$ and $\mathbf{x} \in \ell^2(\mathbb{Z})$ being the vector that contains the input samples $x[n]$.

In general, the smaller the ratio B/A is, the better the numerical properties of the signal model will be. If B/A is close to one, then the assumption of energy preservation may be used without much error when relating the energy of the subband signals $y_m[n]$ to the energy of the input signal $x[n]$ and the output signal $\tilde{x}[n]$. This is important in audio coding applications, as it guarantees that small quantization errors introduced in the subband signals will result in only small reconstruction errors. It enables a bit allocation optimized for minimum error in the subbands to be near-optimal for the final output signal.

The speed of convergence for algorithms like matching pursuit also depends on the frame bounds as shown in Section 3.6. In this context it is to note that the frame realized by a matching pursuit decomposition with a dictionary of atoms $\mathbf{e}_k$ is identical to a frame realized by a filterbank with the matched filters $\mathbf{e}_k^*[-n]$ as shown in Appendix

3.9.1.

The frame operator $\mathbf{S}$ can be represented in the polyphase domain by the $M \times M$ matrix $\mathbf{S}(z) = \tilde{\mathbf{E}}(z)\mathbf{E}(z)$, where $\mathbf{E}(z)$ is the analysis polyphase matrix of the filterbank[122] and the eigenvalues of the frame operator $\mathbf{S}$ equal the eigenvalues $\lambda_n(\theta)$ of the matrix $\mathbf{S}(e^{i\theta}) = \mathbf{E}^H(e^{i\theta})\mathbf{E}(e^{i\theta})$. Bolcskei *et al.*[12] could show that the frame bounds A and B are the essential infimum and supremum, respectively, of the eigenvalues $\lambda_n(\theta)$. Thus, the computation of the frame bounds of overcomplete gammatone signal models using their polyphase matrix representations is possible. Note that in the non-decimated case, the frame bounds and respective eigenvalues are related to the ripple in the overall frequency response of the filterbank.

The eigenanalysis of a signal model is only applicable for a limited frequency interval if the corresponding filterbank is non-decimated. For $N_m > 1$, the mapping of the eigenvalues of the frame operator to the analyzed frequency interval is lost. Thereby the essential infimum and supremum can only be calculated for the entire frequency range, from zero to half the sampling frequency. This results to a lower frame bound of $A = 0$ for bandlimited signal models, like the here analyzed overcomplete gammatone signal model, where filters do not cover frequencies below 20 Hz and above 20 kHz. To circumvent this problem, we added two additional filters for the frequency intervals not covered by the gammatone filterbank, i.e. a lowpass for the $[0, 20]$ Hz frequency interval and a highpass filter for $[20, 48]$ kHz. Thereby we could compute A for a decimated gammatone signal model within the limited frequency range. B was computed without additional filters.

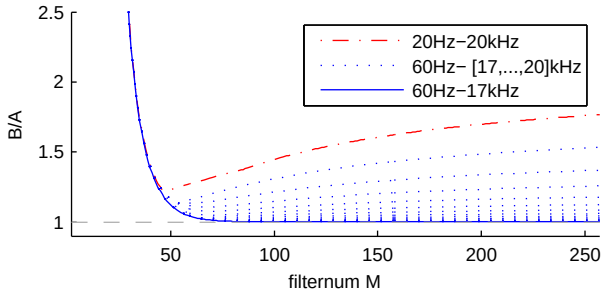


Figure 3.3: (color online) The frame-bound ratios B/A of non-decimated gammatone signal models with the number of filters $M \in [2, 256]$ analyzed over the frequency intervals 20 Hz - 20 kHz and 60 Hz - $[17, 20]$ kHz. For the frequency interval of 60 Hz to 17 kHz, the frame-bound ratio converges towards a tight frame for higher filter numbers.

3.4.2 Analysis of a Non-Decimated Overcomplete Gammatone Signal Model

An overcomplete signal model results in a large quantity of subband coefficients for every filter. To reduce bitcoding and computational costs, it is of interest to know the smallest number M of subbands needed to achieve good frame-bound ratios. As the frame bounds of $\gamma[n]$ are identical with the frame bounds of $\gamma[-n]$, we only need to analyze the frame of the gammatone prototype $\gamma[n]$ itself. The frame bounds A and B of the non-decimated overcomplete gammatone signal model can be computed as described in the previous subsection, and the respective frame-bound ratios B/A are shown in Figure 3.3. The parameters of the analyzed gammatone signal models were $b = 1.019$, $\nu = 4$ with $M \in [2, 256]$ center frequencies between 20 Hz and 20 kHz.

Figure 3.3 shows that the gammatone signal model does not realize a frame for the frequency interval of its center frequencies. The frame-bound ratio is mainly determined by small eigenvalues of the frame operator S found at the first and last gammatone filters (see also Figure 3.11). The ERB scale distributes the center frequencies of the

M	Frequency interval	B	A	B/A	frame
50	[20 Hz, 20 kHz]	1.294	1.046	$B/A = 1.238$	not snug
50	[40 Hz, 17 kHz]	1.294	1.167	$B/A = 1.109$	snug
100	[20 Hz, 20 kHz]	2.462	1.697	$B/A = 1.451$	not snug
100	[60 Hz, 17 kHz]	2.462	2.455	$B/A = 1.003$	$\approx$ tight

Table 3.1: Frame-bound ratios B/A analyzed for different bandlimited signals and number of gammatone filters M.

gammatone atoms in such a way that the overlapping filters result in almost constant eigenvalues. As for the first and the last filters this overlap is not fully realized, the essential infimum of the eigenvalues result in a low lower frame bound A. If we perform the analysis over a reduced frequency interval (see Figure 3.3 and Table 3.1), the frame-bound ratio improves and the gammatone signal is able to achieve a snug frame from $M = 50$ subbands on. This marginal reduction of the frequency interval is non-critical as it still embeds the class of natural sounds with speech for example ranging approximately from 80 Hz to 10 kHz.

For $M = 50$ the frame bounds are $A = 1.167$ and $B = 1.294$, which results in a frame-bound ratio of $B/A = 1.109$. This means that, depending on the actual signal, the energy of the input or output signal of the filterbank may be different from the subband energy by a factor between 1.167 and 1.294. For higher filter numbers the frame-bound ratio converges towards a tight frame and for $M = 100$ a frame-bound ratio of $B/A = 1.003$ is achieved.

For applications that allow a deviation from the human gammatone parameters, we also analyzed the influence of the bandwidth parameters $b \in [0.5, 1.5]$ and the filter orders $\nu \in [4, 20]$ on the frame-bound ratio for the frequency interval from 60 Hz to 17 kHz (see Figure 3.4). For $M = 50$ gammatone atoms, the best frame-bound ratio $B/A = 1.020$ is achieved for a filter order $\nu = 11$ and the bandwidth factor $b = 0.85$.

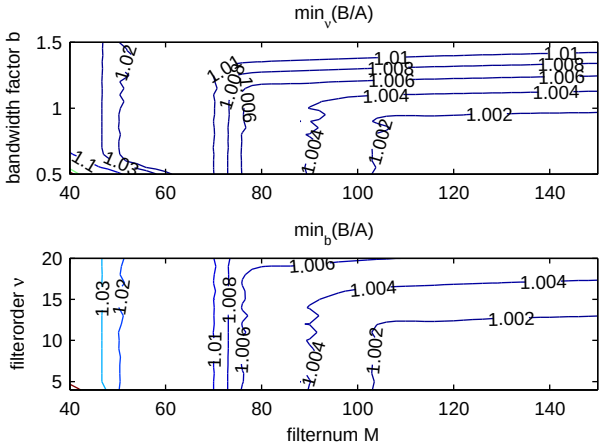


Figure 3.4: (color online) Best possible frame-bound ratios for a fixed bandwidth factor b and filter number M (upper plot) or filter order ν and filter number M (lower plot). The gammatone signal model parameters were $b \in [0.5, 1.5], \nu \in [4, 20], M \in [40, 150]$, analyzed over the frequency interval from 60 Hz to 17 kHz.

For $M = 100$ the filter order $\nu = 12$ and the bandwidth factor $b = 0.5$ result in the lowest frame-bound ratio of $B/A = 1.003$. The contour plot in Figure 3.4 shows that these best frame-bound ratios are located in relatively shallow minima. More generally, we can conclude that for a filter number of $M = 50$, snug frames can be achieved with $b > 0.7$ and all examined filter orders. For $M = 100$ a tight frame is possible with $b \leq 1$, $\nu < 13$. Additionally it can be seen that for a small number of filters ($M < 50$) larger bandwidths achieve better frame-bound ratios. Even more interestingly, for a higher number of filters, large filter bandwidths introduce a decline in the frame-bound ratio which is explained in detail in Section 3.7 and Figure 3.11.

3.4.3 Analysis of a Decimated Overcomplete Gammatone Signal Model

To further reduce encoding and subband processing costs, it is often favorable to remove the redundancy in an overcomplete signal model by downsampling its subband coefficients by factors $N_m > 1$. The decimation of the filterbank coefficients can result in distortions, which will worsen the frame bound ratio of the decimated signal model. Thus, a frame-theoretic analysis can be used to analyze the introduced distortions for different decimation factors N_m. We derived frame bounds for a decimated overcomplete gammatone signal model for the frequency interval from 60 Hz to 17 kHz by introducing additional filters to allow the derivation of A as described in subsection 3.4.1. The resulting frame-bound ratios B/A are shown in Figure 3.5. It can be seen that no snug frame can be achieved for $M \leq 75$ filters with an equal decimation of the subband coefficients. For higher filter numbers, a snug frame can be realized up to an equal decimation of the subband coefficients of $N_m = 4$, $N_m = 5$ and $N_m = 6$ for the filter numbers $M = 100$, $M \in [125, 150]$ and $M \in [175, 255]$, respectively.

To derive optimal decimation factors for an overcomplete gammatone signal model, a full search over all possible N_m by computing the corresponding frame bound ratios would be necessary, which is computational intractable. It is further to note that distortions that fall into a frequency range where the signal has only little energy, will have a minor effect compared to distortions in frequency bands, where most of the signals energy is present. This can not be exploited by an optimization based on framebound ratios due to the lost mapping of the eigenvalues of the frame operator to the analyzed frequency interval. Therefore we introduce and use in the next section an alternative technique to derive optimal decimation factors.

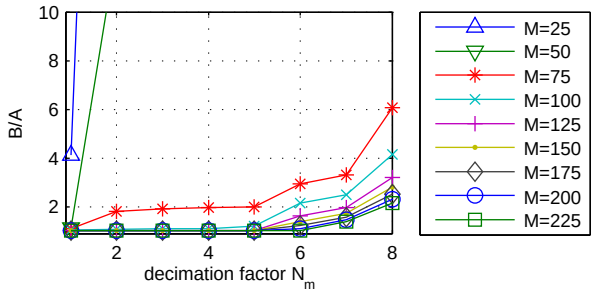


Figure 3.5: (color online) The frame-bound ratios B/A of decimated gammatone signal models with the number of filters $M \in \{25, 50, \cdots, 200, 225\}$ and decimation factors $N_m \in [1, 8]$ analyzed over the frequency interval of 60 Hz to 17 kHz.

3.5 Bifrequency Analysis of a Decimated Overcomplete Gammatone Signal Model

To allow the optimization of decimation factors dependent on the applied signal, we will introduce in this section the bifrequency analysis[126] and define a signal-to-alias ratio (SAR). The bifrequency analysis has the additional advantage that it offers a complete frequency description of the distortions introduced by a decimation of the subband coefficients. This leads to a better insight of the design limitations, i.e. to Condition I and II as given below. This allows to reduce the computational costs of an optimization of the decimation factors. All results in this section were derived with a sampling rate of 44.1 kHz, which is a common sampling rate in signal processing applications like audio coding. The length of the analyzed impulse responses $h_m[n]$ and $g_m[n]$ has been set to 4096 samples or 92.9ms, respectively.

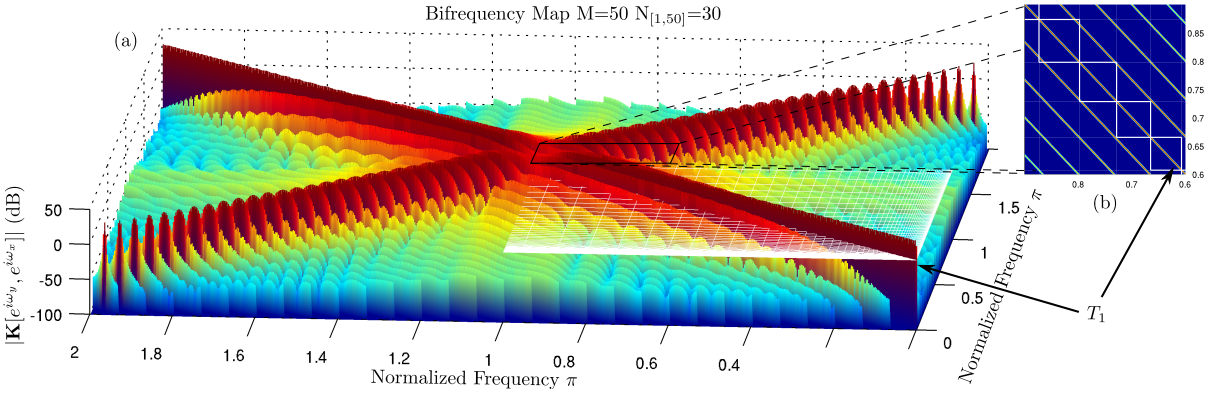


Figure 3.6: (color online) (a) Bifrequency map for a gammatone signal model with the number of filters $M = 50$ and the decimation factor of $N_m = 30$ in every subband. The axes show the normalized frequency domains associated with the input and output signal. The center line represents the time-invariant part (T_1) that maps the input to the output signal and is independent of any decimation. All other lines are due to aliasing terms ($T_{n>1}$) introduced by a decimation of the subband coefficients. The zoom-in (b) shows that in this example in-band aliasing occurs in the last three filters, in which aliasing components fall into the passband of these filters. The filter's passbands are indicated by a grid of thin white lines.

3.5.1 Bifrequency Analysis

An alternative theoretical analysis of the decimated gammatone signal models is possible by the fact that a decimated filterbank can also be understood as a linear time-varying (LTV) system

$$\mathbf{y}[n_y] = \sum_{n_x=-\infty}^{\infty} \mathbf{k}[n_y, n_x]\mathbf{x}[n_x] \tag{3.6}$$

with a periodic system response $\mathbf{k}[n_y, n_x] = \mathbf{k}[n_y + \ell N, n_x + \ell N]$, $\ell \in \mathbb{Z}$, where $\mathbf{x}[n_x]$ is the input and $\mathbf{y}[n_y]$ is the output sequence. $\mathbf{k}[n_y, n_x]$ denotes the response of the system at the discrete time n_y to a unit sample applied at discrete time n_x. For periodic LTV systems, a bifrequency analysis [126] gives a complete description of the system as well

as of its aliasing components. The discrete bifrequency system function[20] is defined as:

$$\mathbf{K}[e^{i\omega_y}, e^{i\omega_x}] := \frac{1}{2\pi} \sum_{n_y=-\infty}^{\infty} \sum_{n_x=-\infty}^{\infty} \mathbf{k}[n_y, n_x] e^{i\omega_x n_x} e^{-i\omega_y n_y} \tag{3.7}$$

relating the input signal spectrum $\mathbf{X}[e^{i\omega_x}]$ to the output signal spectrum $\mathbf{Y}[e^{i\omega_y}]$ with

$$\mathbf{Y}[e^{i\omega_y}] = \int_{-\pi}^{\pi} \mathbf{K}[e^{i\omega_y}, e^{i\omega_x}] \mathbf{X}[e^{i\omega_x}] d\omega_x. \tag{3.8}$$

In the analyzed gammatone signal models, the only periodically time-varying parts are the decimators and interpolators. Therefore, the overall bifrequency map is composed of non-zero unity-slope parallel lines with a constant factor, on whose input and output spectra the effects of the analysis and the synthesis filters, respectively, are projected [83]. The center line represents the time-invariant part of the system, all other lines represent the parts of the system which cause aliasing (see also Figure 3.6). As an objective measure of the aliasing distortions in a signal model we used a signal-to-alias ratio (SAR), defined analogous to the commonly used signal-to-noise ratio. For a given input signal spectrum $\mathbf{X}[e^{i\omega_x}]$ the SAR is defined as

$$SAR(\mathbf{X}[e^{i\omega_x}]) = -10 \log_{10} \left(\frac{T_1^2}{\sum\limits_{n \in \{N_m\}} T_n^2} \right) \tag{3.9}$$

with

$$T_n = \int_{-\pi}^{\pi} \int_{-\pi}^{\pi} \delta(n\omega_x - \omega_y) \mathbf{K}[e^{i\omega_y}, e^{i\omega_x}] \mathbf{X}[e^{i\omega_x}] d\omega_x d\omega_y \tag{3.10}$$

and $\delta(\cdot)$ being the Dirac pulse. The time-invariant part of the system corresponds to T_1, and the aliasing components of the LTV system are represented by the T_n.

To avoid in-band aliasing distortions, N_m must be chosen in such a way that all integer multiples of the decimated Nyquist frequency lie outside the m-th passband of a subband (see Figure 3.6(b)). For an aliasing-free signal model this results in the following necessary condition to prevent in-band aliasing:

Condition I: With ω_m^L and ω_m^H being the starting and stopping cutoff frequencies of the m-th gammatone filter ($0 \leq \omega_m^L \leq \omega_m^H \leq \pi$) it needs to hold:

$$(k\pi/N_m) \notin [\omega_m^L, \omega_m^H] \forall k \in \mathbb{N}. \tag{3.11}$$

This dependency on the bandwidth of the corresponding gammatone filter limits the possible decimation factors to the set which fulfills $N_m < \pi/(\omega_m^H - \omega_m^L)$. In contrast to an ideal bandpass filter, which has a discontinuity in magnitude at the cutoff frequencies, real filters like the gammatone filter exhibit a magnitude response that changes gradually from the passband to the stopbands. A commonly chosen decrease in magnitude to define the cutoff frequency is an attenuation of 3 dB.

Inter-band aliasing can be reduced if the decimation factors are chosen in such a way that an aliasing term of a filter in one subband can be canceled by another aliasing term of a filter in another subband. Such a set of integer decimation factors N_m in which each aliasing term occurs at least twice is called a *compatible set*[52, 30, 122] and needs to fullfil:

Condition II: Let $L := lcm\left(\{N_m\}_{m=0}^{M-1}\right)$ be the least common multiplier (lcm) of the set of decimation factors $\{N_m\}_{m=0}^{M-1}$. If the set is an apposition of repeated distinct integers $\{\mathcal{N}_1, \mathcal{N}_1, \cdots, \mathcal{N}_1, \cdots, \mathcal{N}_{K-1}, \cdots, \mathcal{N}_{K-1}\}$ with $\mathcal{N}_j \in \{N_m\}_{m=0}^{M-1}$ and n_j denoting

the number of $\mathcal{N}_j$ in this set, then it needs to hold:

$$\min \left\{ \frac{lcm(\frac{L}{\mathcal{N}_i}, \frac{L}{\mathcal{N}_j})}{\frac{L}{\mathcal{N}_j}} \right\}_{\substack{i=0 \\ i \neq j}}^{M-1} - 1 < n_j \tag{3.12}$$

3.5.2 Analysis of a Decimated Overcomplete Gammatone Signal Model

We will use the results from the previous subsection to show how optimal decimation factors N_m for a given decimated overcomplete gammatone signal model and a given signal spectrum $\mathbf{X}[e^{i\omega_x}]$ can be derived. Let $\mathbf{N} := (N_0, N_1, \cdots, N_{M-1}) \in [1, M-1]^M$ be the M-dimensional vector space of all possible decimation factors for a gammatone signal model. We can reduce the size of $\mathbf{N}$ by allowing only decimation factors that fulfill Condition I and II. The cutoff frequency was set at 3dB stopband attentuation. The size of the set of possible decimation factors can be further reduced using the constraint $N_0 \geq N_1 \geq \cdots \geq N_{M-1}$, which is derived from Condition I and the fact that the gammatone signal model has monotone increasing bandwidths. To select decimation factors that form a *compatible set*, the decimation factors can be required to be powers of two.

To derive for a given degree of overcompleteness $O = \sum_{m=0}^{M-1} \frac{1}{N_m}$ a set of decimation factors with minimal aliasing distortions, the SAR can be used as a quality measure. To exemplify this, we analyzed an overcomplete gammatone signal model with $M = 50$ filters, center frequencies ranging from 20 Hz to 20 kHz and $N_m \in \{1, 2, 4, 8, 16, 32, 64, 128, 256, 512\}$. We further evaluated if varying the bandwidth of the gammatone filters has an influence on the aliasing distortions. Analyzing Figure 3.6, it can be seen that the major aliasing distortions occur in the high-frequency

bands due to the non-uniform frequency resolution of the gammatone signal model. For applications like speech or audio coding, where only a small amount of signal energy falls in the high-frequency bands, these distortions will have a minor effect compared to the distortions in the low-frequency band, where most of the signal energy is present. Therefore it is favorable to optimize the decimation factors according to the SAR computed for the specific spectrum of the applied signal class. In this example we used the spectrum of the audio test signal "Tom's Diner" by Suzanne Vega (`svega.wav`). Table 3.2 shows the SAR achieved by optimal decimation factors (stated in Appendix B), selected from a set of decimation factors that is constructed as described above and that results in the degrees of overcompleteness $O = 1, 2, \cdots, 8$ respectively. They are compared with commonly chosen decimation factors that are inverse-proportional to the bandwidth of the gammatone filters while fulfilling Condition I. The optimized decimation factors achieve an SAR improvement of 4.7 dB on average compared to the commonly chosen decimation factors that are inverse-proportional to the bandwidth of the gammatone filters while fulfilling Condition I. This can be seen as a significant improvement, recalling that an SAR improvement of 6 dB means a reduction of the distortion energy due to aliasing components by a factor of 2. As the overcomplete

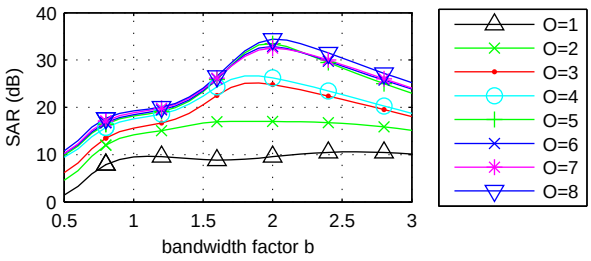


Figure 3.7: (color online) The SAR achieved by optimized decimation factors for a given degree of overcompleteness O and different bandwidth factors b of $M = 50$ gammatone filters.

SAR (dB)	O=1	O=2	O=3	O=4	O=5	O=6	O=7	O=8
optimized N_m	9.5	14.2	15.6	17.5	18.2	18.5	18.9	19.2
prop. bandwidth	6.2	8.1	10.6	11.3	13.4	14.5	14.6	15.7

Table 3.2: The SAR for `svega.wav` and a gammatone signal model with $M = 50$ filters achieved with optimized decimation factors compared to commonly chosen decimation factors that are inverse-proportional to the bandwidth of the filters while fulfilling Condition I.

gammatone signal model realizes for $M = 50$ only a snug frame, we additionally investigated if the SAR can be improved using different filter bandwidths. It showed that for $M = 50$ a deviation from the human bandwidth parameter $b = 1.019$ can reduce inter-band aliasing distortions from 1dB up to 15.2 dB for O=1 and O=8, respectively (see Figure 3.7). As an increase of the filter bandwidth leads to an increase of the energy in the aliasing components, this reduction of aliasing distortions can be addressed to an optimized cancellation of aliasing terms. So depending on the number of applied gammatone filters, the bandwidth factor b should also be included into the optimization process.

3.6 Applications

In this section we report on the signal reconstruction performance of overcomplete gammatone signal models using the example of audio coding and compare the findings with the theoretical results from the previous sections. We applied a coding scheme whose block diagram is shown in Figure 3.2.

In the first experiment, we investigated the signal reconstruction and subband algorithm performance of a non-decimated overcomplete gammatone signal model ($N_m = 1$) as analyzed in Section 3.4. We tested two signal model variations. In the first

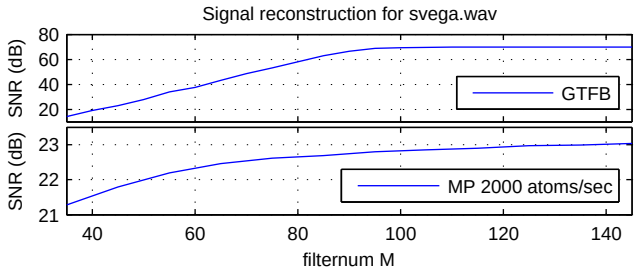


Figure 3.8: (color online) Signal reconstruction experiment using non-decimated overcomplete gammatone signal models for the `svega.wav` test signal. The upper plot shows the results for a signal model without subband processing (GTFB) and the lower plot shows the achieved SNR for a sparse gammatone signal model based on the matching pursuit algorithm (MP).

variation (GTFB), we evaluated the standard overcomplete gammatone signal model with $h_m = \gamma[n], g_m = \gamma[-n]$ and without subband processing. In the second variation, a sparse overcomplete gammatone signal model was realized with $h_m = \gamma[-n]$, $g_m = \gamma[n]$, and a matching pursuit (MP) algorithm [89] was performed in the subband processing block. The stopping condition was set to 2000 atoms per second and it was implemented as described in Appendix 3.9.1. The test signal for this initial audio coding experiment was the commonly used "Tom's Diner" by Suzanne Vega (`svega.wav`). In accordance with the theoretically derived results (Figure 3.3), the signal reconstruction error decreased for both schemes with an increasing number of filters and saturated for higher filter numbers (Figure 3.8). For the overcomplete gammatone signal model (GTFB), near-perfect reconstruction was achieved for $M \geq 100$. For the sparse overcomplete gammatone signal model (MP) the SNR rose to 22.5 dB at $M \approx 70$ and continued to slightly improve further for higher filter numbers until it stayed constant at 23.5dB for $M \geq 500$ gammatone filters. This shows that the convergence speed of the matching pursuit algorithm facilitated also from small frame-bound ratio improvements close to $B/A = 1$, as the overcomplete gammatone signal model did not

contribute further to the signal reconstruction for $M \geq 100$.

We further evaluated a basic perceptual audio coding scheme by scaling the subband coefficients $y_m[n]$ according to a psychoacoustic model before performing a fixed quantization[35]. The psychoacoustic model was realized by the MPEG-2 AAC/MPEG-4 Audio standard reference implementation [63], and a linear 7 Bit quantizer was used. The coding and decoding of the scaled and quantized coefficients was assumed to be lossless and therefore omitted. Finally an according dequantization and rescaling was performed before the audio signal was reconstructed using the synthesis filterbank. We measured the perceived audio quality of the resulting audio signals relative to the original test signal using a model of auditory perception (PEMO-Q) [55]. The estimated perceived audio quality was mapped to a single quality indicator, the Objective Difference Grade (ODG) [66]. This is a continuous scale from 0 for "imperceptible impairment", -1 for "perceptible but not annoying impairment", -2 for "slightly annoying impairment", -3 for "annoying impairment" to -4 for "very annoying impairment". As explained in Section 3.4, subband processing algorithms like perceptual audio coding rely on the assumption of energy preservation in the signal model. Their performance therefore depends on the achieved frame bound ratio of the used signal model. As shown in Figure 3.9, the GTFB signal model without quantization achieved transparent audio coding from $M > 55$ gammatone filters on. Linearly quantizing the subband coefficients to a 7 Bit encoding, the ODG converged around $M > 45$ to approximately -2.5. Scaling the important subband coefficients before quantization according to a psychoacoustic model (PAM) showed an improvement in the perceived audio quality until $M \approx 60$ where an ODG of approximately -1.2 is achieved. With the results from Section 3.4 it can be concluded that for audio coding applications at least a snug frame should be realized by the gammatone signal model. Clearly, to

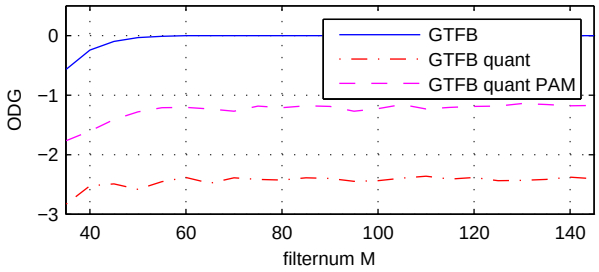


Figure 3.9: (color online) Perceptual reconstruction quality for `svega.wav` encoded without quantization, with a linear quantization and a linear quantization including a psychoacoustic model (PAM).

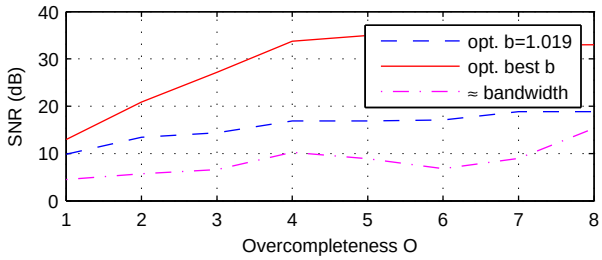


Figure 3.10: (color online) Signal reconstruction experiment using decimated overcomplete gammatone signal models being optimized to maximize the SAR of the test signal (`svega.wav`) as described in Section 3.5.2 compared to commonly chosen decimation factors that are inverse-proportional to the bandwidth of the filters while fulfilling Condition I.

further improve the quality up to an ODG of zero, finer quantization is needed.

In the second experiment, we investigated the signal reconstruction performance of decimated overcomplete signal models with $M = 50$ filters and without any subband processing. As a reference signal model we selected commonly chosen decimation factors that are inverse-proportional to the bandwidth of the gammatone filters, while fulfilling Condition I. We compared their achieved signal reconstruction performance with optimized decimation factors for a gammatone signal model having a fixed bandwidth factor $b = 1.019$ and for a gammatone signal model where also the bandwidth of the filters was optimized as described in Subsection 3.5.2. The audio test file was `svega.wav`

and the results are plotted in Figure 3.10. It can be seen that the decimation factors optimized to maximize the SAR of the audio signal as described in Section 3.5.2 result in a better SNR than the N_m that are increased proportional to the filter bandwidth and fulfill Condition I. It further shows for the snug frame realized with $M = 50$ gammatone filters, a deviation from the human bandwidth parameter $b = 1.019$, if allowed in the context of the application, can reduce the aliasing distortions and improve the signal reconstruction performance.

3.7 Discussion

Applications that use an overcomplete gammatone signal model can be divided into two groups. The first group is concerned with modeling the auditory system. In these studies, the number of auditory filters is inferred from a reasonable filter spacing determined by the estimated bandwidths of the auditory filters. A common value used is one filter per ERB [22, 23, 55], which results in 39 filters for the human cochlea, whose basal end corresponds to 38.9 on the ERB scale [93]. The second group of applications is concerned with signal processing tasks like, for example, audio coding and speech recognition. Hereby, not an accurate replication of the auditory system is strictly needed, but a maximal performance of the algorithm is desired. Therefore, the number of gammatone filters should be chosen optimizing the performance of the subsequent processing algorithms and the introduced computational load. Most signal processing applications using an overcomplete gammatone signal model so far have used psychoacoustically derived filter numbers, which do not result in a frame (see Table 3.3). As show in Section 3.6, subband processing algorithms like matching

pursuit or a perceptual quantizer show an improved performance for improved frame bounds.

Paper	interval of center frequencies	M	given rational	filter per ERB	B/A	frame bound analysis interval	frame
(Ambikairajah *et al.*, 2001)[4]	50 Hz - 7.0 kHz	21	"ripple within 1.5 dB"	—	1.481	100Hz-7kHz	not snug
(Brucke *et al.*,1999)[14]	73 Hz - 6.7 kHz	30	1 filter per ERB	1.0	1.322	70Hz-6.2kHz	not snug
(Feldbauer *et al.*, 2005)[37]	100 Hz - 3.6 kHz	50	frame-bound ratio	2.2	1.003	150Hz-3.0kHz	$\approx$ tight
(Hohmann, 2002)[53]	70 Hz - 6.7 kHz	30	1 filter per ERB	1.0	1.332	65Hz-6.3kHz	not snug
(Kubin *et al.*, 1999)[74]	100 Hz - 3.6 kHz	20	"physiologically-motivated"	0.9	1.364	190Hz-3.1kHz	not snug
(Lin *et al.*, 2001)[81]	$<$ 4 kHz	25	not stated	0.9	1.572	35Hz-4.0kHz	not snug
(Ma *et al.*, 2007)[87]	50 Hz - 8.0 kHz	64	"computational costs"	2.0	1.003	100Hz-6.2kHz	$\approx$ tight
this study	20 Hz - 20.0 kHz	50	frame-bound ratio	1.2	1.109	60Hz-17kHz	snug
		100	frame-bound ratio	2.4	1.003	60Hz-17kHz	$\approx$ tight

Table 3.3: Examples for gammatone signal model parameters found in the literature. The frame bound analysis was performed on a limited frequency interval to exclude distortion effects from the first and last filters.

Note that it is not self-evident that an overcomplete gammatone signal model can achieve a snug frame and converge to a tight frame. The parameters of the gammatone function have been derived from psychoacoustic experiments and are not specifically designed to realize a frame in the mathematical sense. Further analysis of the eigenvalues showed that at higher filter numbers ($M > 60$), the frame-bound ratio is determined mainly by the fact that the frequency spacing of the ERB scale does not fully match the filter overlap to the filter bandwidths. This introduces a positive shift of the largest eigenvalues towards higher frequencies (see Figure 3.11). Therefore

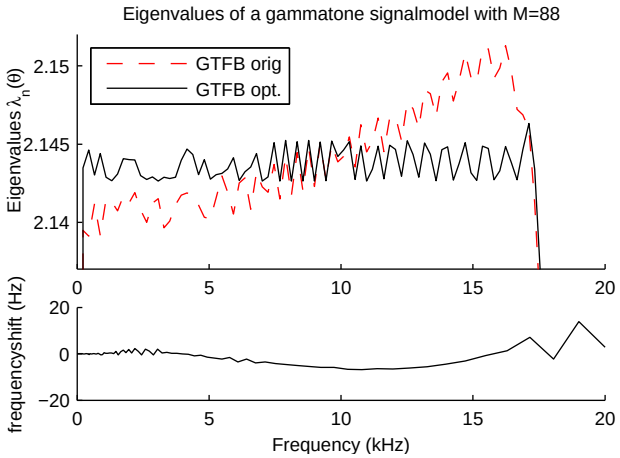


Figure 3.11: (color online) The eigenvalues $\lambda_n(\theta)$ of an overcomplete gammatone signal model with $M = 88$ filters being equally spaced on the ERB scale compared to an optimized frequency scale with frequency shifts applied to the ERB scale as shown in the lower row.

we evaluated if marginal alterations of the filter's center frequency can improve the gammatone signal model. Using the frame-bound ratio as a cost function, a standard optimization algorithm like the Matlab function `fmincon` can be used to derive the frequency shifts necessary to remove the monotonic shift. The derived frequency shifts reduced the center frequencies slightly at middle frequencies, compensating this with a frequency increase at the lower and higher frequencies, see also the example

shown in Figure 3.11. For this example with $M = 88$ the frame-bound ratio could be improved from 1.006 to 1.001 by applying only, relative to the center frequency, marginal frequency shifts. It is to note that these results are only of theoretical interest, as the gammatone signal already forms an almost tight frame at higher filter numbers M, and the derived optimization does not improve the numerical properties of the signal model at a noticeable level. So for the overcomplete gammatone signal model, the ERB scale itself is already close to the frequency tiling of the time-frequency plane that achieves the best frame-bound ratio.

For a decimated overcomplete gammatone signal model, the derived frame bounds can not be used to optimize the decimation factors in dependency of the signal spectrum as explained in Section 3.5. Another possibility to evaluate such bandlimited signal models is the computation of the signal-to-alias ratio, allowing the optimization of the tradeoff between linear amplitude distortions and the amount of aliasing. We could show that the common approach to use decimation factors that are proportional to the bandwidth of the filters is suboptimal. The SAR can easily be computed using a 2D-FFT and we therefore recommend for signal processing applications using a decimated overcomplete gammatone signal model to utilize decimation factors N_m being optimized for the applied signal class.

It is to note that very long finite-impulse responses and high sampling rates have been used in this study to derive frame bounds that are valid approximations for the analog gammatone filters. Applications using other digital realizations of the gammatone filterbank like infinite-impulse response filters might result in slightly different frame bounds [123].

A linear gammatone signal model is a valid approximation of the human auditory filters for moderate sound pressure levels. It has been shown that the filter shape of

the auditory filter changes with stimulus level [110], which led to the development of dynamic, non-linear auditory filter models [84, 56]. The analysis methods applied in this study can not directly be applied to such dynamic filters and are therefore not within the scope of this manuscript.

3.8 Conclusions

Using the theory of frames we could derive that from 2.4 filters per ERB on, a non-decimated overcomplete gammatone signal model achieves near-perfect signal reconstruction and that from $M = 55$ (1.3 Filters per ERB) filters on, a perceptual transparent audio coding is possible. We further showed that by computing a signal-to-alias ratio, the decimation factors in multi-rate signal processing schemes can be optimized, balancing the amplitude and aliasing distortions. We showed for an audio test signal that hereby significant improvements can be achieved.

3.9 Appendix

3.9.1 Matching Pursuit with Matched Filters

Matching Pursuit [89] assumes an additive signal model of the form

$$\mathbf{x} = \sum_{i=1}^{K} s_i \mathbf{a}_i \tag{3.13}$$

with the signal vector $\mathbf{x} \in \mathbb{R}^{N\times 1}$, the coefficients $\mathbf{s} = (s_1, s_2, \ldots, s_K) \in \mathbb{C}^K$, and the atoms $\mathbf{A} = (\mathbf{a}_1, \mathbf{a}_2, \ldots, \mathbf{a}_M) \in \mathbb{C}^{N\times M}$ having unit-norm. For an overcomplete signal model, the matching pursuit algorithm searches for the sparsest encoding in the infinite

number of possible encodings. As mentioned in the introduction, this sparse signal model resembles the signal analysis performed by the human cochlea.

The algorithm performs a greedy iterative search by selecting at the i-th iteration the atom having the largest inner product with the residual $\mathbf{r}_i$:

$$s_{m_i} = \underset{\mathbf{a}_{m_i} \in \mathbf{A}}{\operatorname{argmax}} |\langle \mathbf{r}_i, \mathbf{a}_{m_i} \rangle|^2 \tag{3.14}$$

with m_i being the dictionary index of the selected atom at the i-th iteration. The new residual is then computed with

$$\mathbf{r}_{i+1} = \mathbf{r}_i - s_{m_i} \mathbf{a}_{m_i} \tag{3.15}$$

If we rewrite the inner products in (3.14) as

$$\begin{aligned} s_m &= \langle \mathbf{r}_i, \mathbf{a}_m \rangle \\ &= \sum_{n=1}^{N} r_i[n] \cdot a_m[n] \\ &= \sum_{n=1}^{N} r_i[n] \cdot \tilde{a}_m[N-n+1] \text{ with } \tilde{a}_m[n] = a_m^*[-n] \\ &= r_i[n] * \tilde{a}_m[n] \end{aligned}$$

it can be seen that the inner products can also be computed using the time reversed atom $\tilde{a}_m$, which is also called a *matched filter*. So we can efficiently compute all inner products using a time-reversed gammatone filterbank. In practical applications of matching pursuit the support L of the atoms is often much smaller than the length N of the signal. Therefore most implementations [88, 38, 50] divide the signal into

overlapping blocks of length L and stepwidth S. With this iterative procedure, only the correlations of the $2L/S - 1$ signal blocks which have been altered in the previous iteration need to be recomputed. Using the matched-filter approach we can compute the new correlations of the $2L/S - 1$ signal blocks in one step by convolving the $2L$ samples of the whole block once with the matched filterbank. So for a signal of length N and a dictionary size M, we can perform the matching pursuit iteration in $\mathcal{O}(MN)$. If matching pursuit is performed with a pure gammatone dictionary, we can accelerate the matching pursuit algorithm further by precomputing the representations of the gammatone atoms in the filterbank domain and performing the update of the inner products by a simple subtraction in the filterbank domain. For a dictionary of size M, instead of $6M \cdot 2L$ multiplication and $10M \cdot 2L$ additions[51], the update of the correlations can be done with $M2L$ subtractions.

3.9.2 Optimal Decimation Factors

The in Section 3.5.2 derived optimal decimation factors for `svega.wav`, $b = 1.019$ and $M = 50$ are:

$O = 1$ $N_{1-10} = 128, N_{11-33} = 64, N_{34-49} = 32, N_{50} = 16$

$O = 2$ $N_{1-8} = 64, N_{9-36} = 32, N_{37-48} = 16, N_{49-50} = 8$

$O = 3$ $N_{1-24} = 32, N_{25-40} = 16, N_{41-50} = 8$

$O = 4$ $N_{1-10} = 32, N_{11-31} = 16, N_{32-50} = 8$

$O = 5$ $N_{1-2} = 32, N_{3-31} = 16, N_{32-44} = 8, N_{45-50} = 4$

$O = 6$ $N_{1-20} = 16, N_{21-44} = 8, N_{45-49} = 4, N_{50} = 2$

$O = 7$ $N_{1-14} = 16, N_{15-39} = 8, N_{40-49} = 4, N_{50} = 2$

$O = 8$ $N_{1-10} = 16, N_{11-39} = 8, N_{40-46} = 4, N_{47-50} = 2$

Acknowledgments

The authors would like to thank the anonymous reviewers for their constructive comments and corrections, which significantly improved the quality of this manuscript. This work was partly funded by the German Science Foundation (DFG) through the International Graduate School for Neurosensory Science and Systems and the SFB/TRR 31: 'The Active Auditory System'. The author Stefan Strahl wants to especially thank Astrid Klinge for the inspiring scientific discussions about the manuscript and wants to ask her to marry him.

4 An Adaptive Tree-Based Progressive Audio Compression Scheme[1]

4.1 Abstract

A fine-grain scalable and efficient audio compression scheme based on adaptive significance-trees is presented. Common approaches for 2-D image compression like EZW (embedded wavelet zero tree) and SPIHT (set partitioning in hierarchical trees) use a fixed significance-tree that captures well the inter- and intraband correlations of wavelet coefficients. For 1-D audio signals, such rigid coefficient correlations are not present. We address this problem by dynamically selecting an optimal significance-tree for the actual audio frame from a given set of possible trees. Experimental results are given, showing that this coding scheme outperforms single-type tree coding schemes and performs comparable to the MPEG AAC coder while additionally achieving fine-grain scalability.

4.2 Introduction

Recent advances in wireless audio streaming ([10],[5]) and the increase of heterogeneous networks like the Internet introduced problems such as bitrate fluctuation, different

[1]This chapter in the present form has been published in Proceedings of the IEEE WASPAA05, New Paltz, USA, pp. 219-222 (2005).

target channel capacities or storage costs for multi-bitrate files. Storing the data in an embedded manner can address this issue in a generic manner.

Bitplane coding and significance-trees have been successfully applied to image coding ([113],[111]). Such coding schemes successfully capture the structure of the wavelet-based image representation, making very efficient sorting passes and a low number of sorting bits possible. Such natural rigid correlations cannot be found in audio signal representations like the MDCT transform, necessitating the derivation of optimal significance-trees in a data dependent manner.

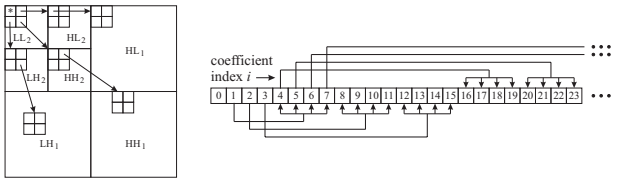
How to generate these significance-trees capturing the variant spectral distribution of audio data and the principle of our progressive compression scheme, called combined significance-tree quantization (CSTQ) using these significance-trees, are discussed in Section 4.3. In Section 4.4, we present experimental results on audio compression including subjective listening tests.

4.3 Basic Concepts

4.3.1 Significance-Trees

Significance-tree coding algorithms like EZW [113] or SPIHT [111] exploit the fact that it can be beneficial to describe significant coefficients of a bitplane via their position and value information instead of transmitting all values one by one. These spatial orientation trees can be mathematically represented using parent-children coefficient coordinate relationships. Fig. 4.1a shows the case of image compression, were the offspring $O(i,j)$ of the wavelet parent coefficients at position (i,j), except for the highest and lowest pyramid level, have been defined as $O(i,j) = \{(2i,2j),$

$(2i, 2j+1), (2i+1, 2j), (2i+1, 2j+1)\}$. Due to the fact that the 2-dimensional wavelet transformation has a typical coefficient inter- and intra-band correlation [82], this rigid tree structure can capture the correlation with a reasonable computational complexity, giving an efficient compression scheme.



(a) (b)

Figure 4.1: Parent-offspring dependencies in SPIHT with different styles. (a) 2-D tree. (b) 1-D tree following the offspring rule $O(i) = iN + \{0, N-1\}$.

For 1-dimensional audio signals, the problem of selecting the optimal tree structures remains unsolved despite considerable efforts. Most existing algorithms use a single type of tree as shown in Fig. 4.1b with the fixed parent-children relationship $O(i) = iN + \{0, 1, \cdots, N-1\}$ for different positive integers N. For the MDCT transform, $N = 4$ was adopted in [34, 107, 106, 108] and the wavelet packet transform was encoded using $N = 2$ in [85, 86]. This type of tree will be referenced in the following as SPIHT-style significance trees.

4.3.2 Bitplane coding using Significance-Trees

The set of M transform coefficients to be encoded for an audio frame is denoted by the vector $\mathcal{X} = (X_1, X_2, \ldots, X_M)$, and the according coordinates set is denoted by $\mathcal{M} = (1, 2, \cdots, M)$. The algorithm starts with the most significant bitplane n_{max},

which can be easily computed with $n_{max} = \lfloor \log_2(\max_{i \in \mathcal{M}}\{| X_i |\}) \rfloor$. A coefficient X_i can then be expressed as

$$X_i = s \sum_{k=n_{min}}^{n_{max}} b_{i,k} 2^k$$

with $b_{i,k} \in \{0, 1\}$ and $s \in \{\pm 1\}$ being the sign. If X_i is an integer value, then $n_{min} = 0$. To encode real-valued coefficients, n_{min} can be negative.

During the bitplane-coding process, all bitplanes $n \leq n_{max}$ are processed iteratively (i.e., the bits $b_{i,n}$, $i = 1, 2, \ldots, M$ are transmitted) in so-called sorting and refinement passes [111]. In a sorting pass, all coefficients that become significant with respect to the actual bitplane n are found by employing tests on the coefficient absolute values, and these test results are written to the output bitstream. For coefficients that are found to be significant, also a sign bit is transmitted. During the refinement passes, the lower bitplanes of already identified significant coefficients are transmitted.

The sequence of the coefficient sorting is defined by the significance-tree so that all elements in the coefficient set $\mathcal{X}$ are uniquely mapped into nodes in the trees. Each significance tree $\mathcal{T}$ is composed of several nodes that link coefficient coordinates i (position information) of scalars X_i in a hierarchical manner. A tree $\mathcal{T}$ is said to be significant with respect to bitplane n if any scalar inside the tree is significant, that is, if the magnitude of at least one coefficient in the set is larger than 2^n. The pseudocode of the sorting pass is as follows:

TreeSignificance (current tree $\mathcal{T}$, current threshold 2^n)

- *If $\mathcal{T}$ is insignificant with respect to 2^n, emit '0' and return;*
- *If $\mathcal{T}$ is significant with respect to 2^n, emit '1';*

- *If root node $N(\mathcal{T})$ is significant with respect to 2^n, emit '1', otherwise emit '0';*
- *Call TreeSignificance() for each subtree with root node as offspring of $N(\mathcal{T})$ with threshold 2^n;*
- *Return;*

4.3.3 Proposed Adaptive Significance-Tree Selection

The SPIHT-style significance trees proposed for audio coding so far are rather arbitrary. They are simply derived by projecting the known 2-D trees into the vector notation of 1-D structures. To establish better tree structures and to capture the dynamically variant spectral behavior of audio signals, we predefine a set of significance-trees and dynamically select the locally optimal ones for each audio frame.

For tree construction, in general, it is important to recall that trees should be built in such a way that the coefficients that are most likely to be large in magnitude are located close to the roots of the trees, whereas the small coefficients should be located at the outer leaves. The larger the (sub)-trees that contain small coefficients are, the more efficient the coding will be. In contrast to [127] we used non-complete significance trees by placing remaining nodes at the last treelevel.

In this paper we design the set of μ possible significance-trees by constructing these trees out of m subtrees with different roots and different sorting orders. The coding cost to encode the tree selection information is $log_2(\mu)$ bits per frame. We considered $m = 8$ with equally sized subtrees and $m = 10$ with logarithmically sized subtrees. See Fig. 4.2 for an illustration of the trees. Each subtree was selected from four different types of trees (ascending, descending, concave oder convex) yielding $\mu = 65.536$ possible trees

(tree selection needs 16bit per frame) for the equally sized and $\mu = 1.048.576$ (bit cost of 20bit per frame) for the logarithmically sized subtrees.

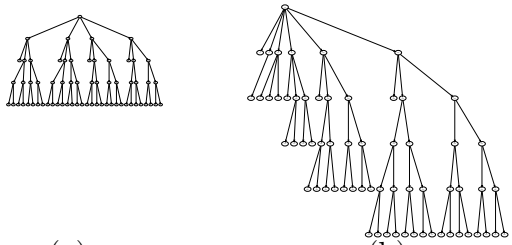


Figure 4.2: Examples of possible significance-trees with treeorder $N = 2$ and frame-length $M = 64$ (a) $m = 4$ (equally sized trees). (b) $m = 6$ (log-sized trees).

For a given audio frame to be encoded, we select the tree that allows us to encode the largest number of high-magnitude coefficients within the first ν tree levels. In the experiments, ν was set to 3.

4.3.4 CSTQ Algorithm

Let us assume that a set of optimal local significance trees for transmitting a coefficient set $\mathcal{X}$ has been found, for example, through testing the efficiencies of various possible trees as mentioned above. The compression scheme then operates as follows: Iteratively, all bitplanes $n = n_{max}, n_{max} - 1, n_{max} - 2, \ldots, n_{min}$ are processed in sorting and refinement passes. In a sorting pass, all coefficients that become for the first time significant (i.e., their magnitude exceeds the current threshold 2^n) are logged to a list of significant coefficients (LSC) and their signs are encoded. This means, at any point in the encoding process, the LSC contains the coordinates of all coefficients that have

been found to exceed the current test threshold of 2^n. When all significant coefficients with respect to the current threshold 2^n have been identified and their coordinates have been moved to the LSC, the refinement pass stores the bitplane information for the significant coefficients by processing the LSC, except for the coefficients that were included in the last sorting pass. The overall algorithm is as follows.

CSTQ Algorithm:

1. *Tree Generation:* select one of the μ possible significance-trees, containing m local subtrees;

2. *Initialization:* output $n = \lfloor \log_2(\max_{i \in \mathcal{M}}\{\mid X_i \mid\}) \rfloor$; output selected significance-tree; sequentially do: set LSC (list of significant coefficients) as an empty list.

3. *Sorting Pass:* sequentially call *TreeSignificance*, move all significant coefficients into the according LSC, output their signs.

4. *Refinement Pass:* sequentially, for each coefficient in according LSCs, except those included in the last sorting pass, output the n^{th} most significant bit of X_i.

5. *Quantization-Step Update:* decrement n by 1 and go to Step 3.

The process is repeated until the desired bit budget is achieved, or, in case of lossless compression, all bits in all coefficients have been encoded.

4.4 Experimental Results

4.4.1 Comparison of significant tree models

In this section, we compare the performance of adaptively selected and fixed significance trees. The number of possible trees for our algorithm was set to $\mu = 65.536$ (equally sized) and $\mu = 1.048.576$ (logarithmically sized), respectively, as described in Section 4.3.3.

The audio signal was selected as the `cha2.wav` file [62] (mono, 16 bits, 48 kHz) and the bitrate was set to $R = 96$ kbps. A MDCT filterbank was used to remove the signal redundancy and the framesize was set to $M = 1024$. The frame bit budget R_f was computed as $R_f = \lfloor R \cdot M/Fs \rfloor$ where Fs is the sampling rate in Hz, yielding $R_f = 2048$ bits per frame for 96 kbps. The treeorder of the significance trees has been set to $N = 4$. As a quality measure, the frame-wise signal-to-noise ratio (SNR) was used, which was computed as the ratio of a frame's energy, divided by the energy of the reconstruction error in the frame. The two scenarios gave the results listed in Table 4.1.

scenario	SPIHT	CSTQ linear spaced	CSTQ log-spaced
segSNR	32.99	34.27	34.56

Table 4.1: Average frame-wise SNRs in dB for the `cha2.wav` signal coded at 96 kbps, using different algorithms.

From Table 4.1 it can be seen that an adaptive significance-tree selection benefits from the variant spectral distribution of audio data and that a logarithmic spacing, similar to the one that can be found in the human auditory system, is a good strategy to exploit the structure of audio signals.

4.4.2 Combination with the MPEG AAC Standard

In this experiment, we use the state-of-the-art MPEG AAC compression scheme and combine it with our CSTQ algorithm in order to achieve progressive coding. For this, we keep the AAC scheme unchanged up to the point where Huffman coding is employed, then apply the CSTQ algorithm to realize the compression of the quantizer indices. In all experiments, the reference software of [60] was used.

The compression of quantizer indices can either be lossless or lossy, depending on the number of bits transmitted. On the decoder side, the received quantizer indices (either exact values or approximations, depending on the bitrate) are injected into the standard AAC decoder. All other side information is transmitted as produced by the AAC coder.

Table 4.2 shows the average segmental SNRs for the algorithms at different bitrates, using signals from the sound quality access material (SQAM) [36] and from the 1990 MPEG evaluation [62]. Note that the results for the AAC coder were produced by encoding the signal individually for each bitrate. For CSTQ, the encoding was done once at 64 kbps, and then lower rates were realized by truncating the frame-wise embedded bitstream produced by the CSTQ algorithm. As the results in Table 4.2 show, the SNR for CSTQ is slightly lower at the highest bitrate, but it is better for all lower bitrates. A similar behavior could be found for other audio material as well. This could be explained by to the fact that at 64 kbps, not all frames could be compressed by the CSTQ scheme in a lossless manner within the given bit budget. At lower rates, however, CSTQ has the advantage that it can exactly meet the target bitrate without the need of including any padding bits, which are quite common in the AAC bitstream produced by the reference software.

audio file	Time (min)	Bitrate (kbps)	AAC	AAC BSAC	AAC CSTQ linearly spaced $N = 2$	AAC CSTQ log. spaced $N = 2$
Tracy Chapman [62]	0:37	16	10.46	1.26	7.65	8.02
		24	12.90	8.25	9.57	10.32
		32	14.19	12.08	13.63	13.87
		40	15.04	14.04	14.89	14.96
		48	15.59	14.94	15.43	15.49
		56	16.09	15.51	16.01	16.03
		64	16.47	15.54	16.43	16.44
female English speech [36]	0:21	16	7.65	5.59	9.74	9.98
		24	10.48	10.03	12.51	13.30
		32	12.54	12.66	15.50	15.74
		40	13.70	15.53	18.07	18.12
		48	15.28	16.97	19.36	19.41
		56	16.75	17.05	19.89	19.91
		64	19.98	17.07	19.98	19.98
quartet [36]	0:28	16	7.42	6.03	8.51	8.95
		24	9.59	9.48	10.67	11.03
		32	11.32	11.43	13.37	13.51
		40	12.73	13.89	15.23	15.30
		48	14.29	14.77	16.32	16.36
		56	15.82	14.95	16.84	16.86
		64	17.05	14.95	17.03	17.03

Table 4.2: Average segmental SNRs in dB for different signals, bitrates and algorithms

4.4.3 Subjective Listening Tests

In order to see whether the objective results based on the segmental SNR translate into similar subjective quality impressions, we carried out listening tests with twenty test persons for the scenario with eight equally sized subtrees per frame. In these tests, the CSTQ-scheme was compared with the MPEG2-AAC standard and the MPEG-4-AAC-BSAC standard, which is currently the only standardized fine-grain progressive

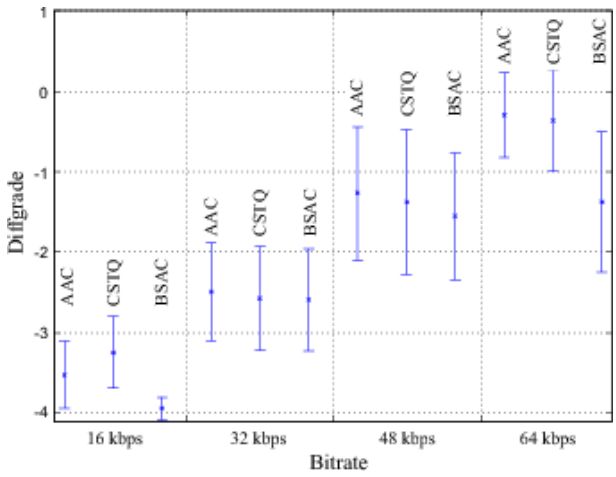


Figure 4.3: *Subjective difference grades for different codecs at bitrates between 16 and 64 kbps for one mono channel.*

audio compression scheme. Also for MPEG-4-AAC-BSAC, the reference software from [60] was used. The measurement procedure was set up according to the ITU recommendation BS.1116-1 [65]. The quality ratings between one (very annoying) and five (indistinguishable from the original) were translated into the subjective difference grade, which is the difference between the rating for the encoded test item and the hidden reference and ranges from zero (equal quality) down to -4 (the lowest grade). The results for three different test signals are depicted in Fig. 4.3. As one can see, the performance of CSTQ is almost equal to the AAC performance, and it is significantly better than the BSAC one.

4.5 Conclusions

The fine-grain scalable audio signal compression problem has been addressed in this study. While in almost all existing algorithms, a single type of significance-tree has been adopted for sorting significant coefficients and transmitting position information,

we have proposed a novel adaptive significance-tree technique. Such a tree is generated dynamically to suit variant spectral behavior from frame to frame. It could be shown that a logarithmic tree size scaling captures better the harmonic structure of an audio signal. Based on the dynamic tree selection, a compression scheme called CSTQ has been proposed, which provides both high compression quality and fine-grain bitrate scalability. Experimental results clearly demonstrate the following properties: the method outperforms the existing SPIHT-like algorithms and yields competitive quality as the nonscalable AAC audio compression scheme, yet with fine scalability of one-bit granularity per frame.

5 A Dynamic Fine-Grain Scalable Compression Scheme with Application to Progressive Audio Coding[1]

5.1 Abstract

This paper studies the fine-grain scalable compression problem with emphasis on 1-D signals such as audio signals. Like in the successful 2-D still image compression techniques EZW (embedded zerotree wavelet coder) and SPIHT (set partitioning in hierarchical trees), the desired fine-granular scalability and high coding efficiency are benefited from a tree-based significance mapping technique. A significance tree serves to quickly locate and efficiently encode the important coefficients in the transform domain. The aim of this paper is to find such suitable significance trees for compressing dynamically variant 1-D signals. The proposed solution is a novel dynamic significance tree (DST) where, unlike in existing solutions with a single type of tree, a significance tree is chosen dynamically out of a set of trees by taking into account the actual coefficients distribution. We show how a set of possible DSTs can be derived that is optimized for a given (training) dataset. The method outperforms the existing scheme for lossy audio compression based on a single-type tree (SPIHT) and the scalable audio coding schemes MPEG-4 BSAC and MPEG-4 SLS. For bitrates less than 32 kbps it results in an improved perceived audio quality compared to the fixed-bitrate

[1]This chapter is in print in the present form in the IEEE Transaction on Audio, Speech, and Language Processing (2010).

MPEG-2/4 AAC audio coding scheme while providing progressive transmission and finer scalability.

5.2 Introduction

The main attractive feature of a scalable compression algorithm is the possibility of progressive transmission. Scalability enables the receiver to decode the received signal at different fidelity levels, depending on the actual needs and available device capabilities, and it also allows one to adapt the data rate to the actual channel capacity. Certainly, such a feature is extremely desirable in packet-based networks, and it has been exploited to handle problems such as data-rate fluctuation, channel congestion or limited storage space.

For audio compression, the majority of classic encoders optimize on a single (although arbitrary) target compression ratio, striving to deliver the best quality given the bitstream length, or to deliver the shortest bitstream length given a constraint on quality. An example is the well-known MPEG2/MPEG4 advanced audio coder (MPEG-2/4 AAC) [13], which is a state-of-the-art audio compression tool that provides excellent quality at bitrates of 64 kbps (kbit per second) per channel and also yields excellent performance, relative to the alternatives, at bitrates reaching as low as 16 kbps. Clearly, no scalability is offered from those conventional coders.

To cater for the scalability desire, a few scalable encoders that are organized in layers have been proposed and standardized. Namely, ITU-T G.727 [69] (5-, 4-, 3- and 2-bit/sample) for telephone bandwidth or ITU-T G.722 [68] (48/56/64 kbps) for wideband speech. More recently, MPEG-4 CELP [63] (2 kbps in the narrowband version and 4 kbps in wideband) and MPEG-4 BSAC [63] (with up to 1 kbps fine

bitrate graduation) have been introduced. BSAC stands for bit slice arithmetic coding. In the MPEG-4 BSAC coder, a core layer produces the lowest bitrate and provides the minimum information to obtain a basic quality for the decoded signal, and several enhancement layers contain additional information that allows the decoder to improve the quality. The scalability is obtained by transmitting the core layer bitstream, combined with one or more enhancement layers. Clearly, the obtained granularity relies on the pre-defined bitrates allocated to the layers and the number of layers sent to the decoder. Typically, for achieving scalable bitrates between 16 and 64 kbps, up to 48 enhancement layers are used. The newest scalable audio coding scheme addition to the MPEG-4 standard is MPEG-4 SLS [64] which provides backward compatibility to MPEG-2/4 AAC while achieving a granularity of up to 0.4 kbps. SLS stands for scalable to lossless audio coding and the MPEG-4 SLS coder achieves scalability by using, similar to the MPEG-4 BSAC coding standard, a layered concept. The bitstream consists of an AAC core layer defining the lowest possible bitrate and a lossless enhanced (LLE) layer that produced the fine grain scalable to lossless portion of the lossless SLS bitstream. The LLE layer encodes the residual signal using a bitplane coding approach whose quantizer noise reproduces the perceptually optimized spectral shape of the AAC core layer's quantizer noise [125]. The MPEG-4 SLS standard also defines a computational less complex non-core mode for applications that require only lossless quality.

In all existing standardized scalable encoders, the scalability is obtained at the price of degradation in terms of performance when compared to fixed-rate schemes, and, in general, the finer the granularity is, the higher the loss is [71]. A few other, non-standard coders with fine bitrate scalability have been proposed as well. These are, for example, coders based on the SPIHT principle [34, 107, 106, 108, 85, 86], which

will be discussed in more detail below, the embedded wideband speech coder in [109], which is based on layered multimode transform predictive coding, the layered coder in [71], which orders and transmits parameters in terms of their significance, and the embedded audio coder in [78], which forms embedded streams for time segments of about 0.74 seconds duration. For the MPEG-4 SLS scheme, a prioritized bitplane coding has been proposed [79], that similar to the approach presented in this work, divides the frequency spectrum of the LLE layer into several regions and assigns these regions with coding priorities according to their respective energy levels.

Several of the previously mentioned scalable coders use bitplane coding (also called bit-slice coding), where the coefficients are transmitted layer by layer, starting with the layer of most significant bits. In the first encoding round, this provides coarse representations of the largest coefficients (largest in magnitude), and subsequent layers provide more accurate representations of the coefficients. For a larger set of coefficients to be encoded, in order to achieve efficient bitplane coding, it is advantageous to describe the bitplanes via position and value information, instead of transmitting value information alone in a straightforward manner. This is in particular interesting for sparse data where most of the coefficients are zero. One successful solution of this is the tree-based significance mapping technique [113, 111]. In these methods, for a set of coefficients to be encoded, by assuming a known coefficient significance/magnitude distribution in the form of trees, the coefficient position information is mapped into node-location information in the tree domain. Moreover, different significance-tree structures result in different compression efficiencies. Details will be given in Section 5.3.

For 2-D coefficient-set compression, applying tree-based techniques has produced impressive advances in wavelet-based image compression. Its development could be

traced back to the work of Lewis and Knowles [77], who used tree structures to exploit the statistical properties found in the pyramidal decomposition of natural images. The technique was further developed by Shapiro [113], who proposed an efficient method to combine the two techniques, bitplane coding and tree-based significant-coefficient selection (sorting), and applied them to the wavelet transform coefficients. This combination, called embedded zerotree wavelet (EZW) algorithm, was later refined by Said and Pearlman [111]. The according new algorithm, called set partitioning in hierarchical trees (SPIHT), is one of the state-of-the-art progressive image compression algorithms. It has been realized that the reason for the success of the SPIHT fine bitstream scalability, state-of-the-art compression performance, and reasonable computational complexity, is mainly attributed to the effective description of the significance map of wavelet coefficients. This has been confirmed from both empirical observations and theoretical analysis: in the experiment illustrated in [82], the SPIHT algorithm successfully captures not only the inter-band correlation but also the intra-band correlation. Theoretical support can be found in [114] and [80], where the statistical models between the magnitudes of wavelet coefficients in different scales and orientations were proved to be existent.

Inspired by the success in image compression, SPIHT-related coding techniques have been proposed for audio compression as well [34, 107, 106, 108, 85, 86]. In all of these approaches, the tree structures have been fixed independent of the signals to be encoded and are based on the assumption that low-frequency components contain more energy than high-frequency ones. This assumption, however, does not hold for all frames of real-world audio signals, so that fixed trees can only be suboptimal.

To address this problem, in an earlier work, the authors of this paper proposed an adaptive tree-based significance mapping technique that used a fixed set of significance

trees from which the optimal tree for each frame was selected [127, 120]. In the present paper, we present a novel, scalable compression scheme with a dynamic set of data-dependent significance trees, called dynamic significance tree quantization (DSTQ). As the compression performance of a tree depends on how well it matches the signal structure, the set of significance trees should be optimized for the applied signal class. We propose to derive such a set of data-dependent significance trees directly from the coefficient's distribution. An initial set of significance trees can be either learned for a general signal class (e.g., speech), or they can be optimized for the specific signal to be encoded. A dynamic adaptation of the set of trees is possible online without sending further side information if the encoder and decoder stage use the same learning algorithm. The proposed DSTQ algorithm can provide bitrate scalability at a granularity of one bit per frame, and has better performance than the existing SPIHT-related algorithms.

Another concept of scalable audio coding is the optimization at the encoding stage for a wide range of bitrates and types of input signals for an a priori known bitrate. For this problem, hybrid sound coding schemes have been proposed [112], using in parallel sinusoidal, transform or CELP coding modules and optimizing the respective bitrates or time segmentation using for example operational rate-distortion optimization. These scalable audio coders result in a bitstream with a fixed bitrate and are therefore different from the scalable audio coders like MPEG-4 BSAC, MPEG-4 SLS or our proposed coding schemes, where the bitrate can be changed after the encoding process, scaling only the bitstream itself.

This paper is organized as follows. To facilitate the later description of our algorithm, a brief overview of existing SPIHT-style algorithms (both in image and audio compression) is presented in the next section. Then, Section 5.4 describes our proposed

dynamic significance tree method and develops the scalable compression scheme DSTQ. We present a data-driven approach to generate a set of optimal significance trees. To illustrate the compression performance, we compare in Section 5.5 the achieved signal reconstruction performance with the SPIHT scheme for lossy audio compression based on a single-type tree and our previously proposed scheme using a fixed set of significance trees (CSTQ). We further evaluate the achieved perceptual quality for compression of quantizer indices compared with the fixed-bitrate MPEG-2/4 AAC and scalable MPEG-4 BSAC and MPEG-4 SLS audio coding schemes. Conclusions are given in Section 5.6.

Notation

Matrices and vectors are printed in boldface, sets are printed in script alphabet and trees in fraktur alphabet. $\oplus$ denotes the addition of sets and $\mathbb{N}^+$ is the set of all positive integers excluding zero. $\lfloor \boldsymbol{x} \rfloor$ denotes the greatest positive integer less than or equal to a given positive real number $\boldsymbol{x}$. $[\boldsymbol{a}, \boldsymbol{b}] := \{\boldsymbol{x} \in \mathbb{N}^+ | \boldsymbol{a} \leq \boldsymbol{x} \leq \boldsymbol{b}\}$ represents the set of all positive integers between and including $\boldsymbol{a}$ and $\boldsymbol{b}$.

5.3 Tree-based Significance Mapping in SPIHT

5.3.1 The SPIHT Algorithm in Image Compression

In this section, we give a brief summary of some characteristics of the SPIHT algorithm, introduced by Said and Pearlman in [111] for 2-D wavelet-based image compression.

Assume an original image is wavelet transformed to a 2-D coefficient array $\mathbf{X}$. Each element $\boldsymbol{X}_{(i,j)}$ of $\mathbf{X}$ is called transform coefficient at coordinate $(\boldsymbol{i}, \boldsymbol{j})$ and represented

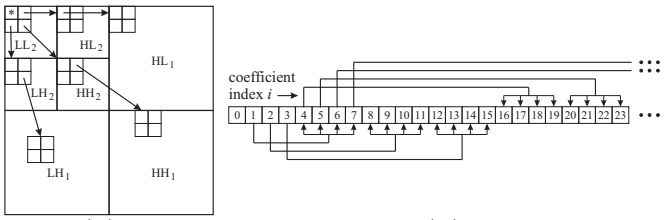
in its binary form. Note that for efficient information transmission, the most significant value information should be transmitted first. To achieve this, the idea is to encode the coefficient value information in decreasing bitplane order. In particular, the sign and value of a coefficient are encoded only when the coefficient has become significant, that is when its most significant nonzero bit is located at the current or one of the previous bitplanes. This idea leads to another issue, which is the question of how to efficiently encode the coordinates of significant coefficients. A good solution is provided by the tree-based significance mapping technique.

A tree is a set of linked nodes that realizes a hierarchical data structure. Each node has at most one parent node and a set of zero or more children nodes. A node with zero child nodes is also called a leaf node and the root node is defined as the topmost node that has no parent node. The order of the tree defines the number of children of a node. A significance tree is generated by ordering all coefficients in the form of trees with the assumption that the coefficients closer to the roots of the trees will usually be more significant (i.e., larger in magnitude) than those at the leaves. In SPIHT coding of the wavelet coefficients of an image, such a tree is recursively generated by the parent-offspring relationship $\boldsymbol{\mathcal{O}(i,j) = \{(2i,2j),(2i,2j+1),(2i+1,2j),(2i+1,2j+1)\}}$, where $\boldsymbol{(i,j)}$ is the coordinate of a parent and $\boldsymbol{\mathcal{O}(i,j)}$ the set of coordinates of its offspring (also called direct descendants of the parent at node $\boldsymbol{(i,j)}$). Each of the members of the set $\boldsymbol{\mathcal{O}(i,j)}$ then has its own offspring, which are called indirect descendants with regard to the parent node $\boldsymbol{(i,j)}$. For the description of the algorithm, all the indirect descendants of the parent with coordinates $\boldsymbol{(i,j)}$ are gathered in the set $\boldsymbol{\mathcal{L}(i,j)}$. Finally, a complete descendants set $\boldsymbol{\mathcal{D}(i,j)}$ is defined as the sum of the direct and indirect descendants $\boldsymbol{\mathcal{D}(i,j) = \mathcal{O}(i,j) \oplus \mathcal{L}(i,j)}$.

The above mentioned relationship between parents and their offspring provides a

natural link between wavelet coefficients in different frequency bands that belong to the same spatial location in an image. It holds for all coefficients, except the ones in the highest and lowest frequency bands, because the coefficients in the highest bands do not have any offspring, and in the lowest band, only three out of four coefficients have offspring. Fig. 5.1(a) gives an illustration.

Based on the significance tree, which is also called spatial orientation tree, the entire SPIHT algorithm performs iterative sorting and refinement passes to progressively encode the coefficient array $\mathbf{X}$. From the top bitplane $\boldsymbol{n_{max}} \in \mathbb{N}^+$, naturally decided by $\mathbf{2}^{\boldsymbol{n_{max}}} > \max_{\boldsymbol{(i,j)}}(|\boldsymbol{X_{(i,j)}}|/\mathbf{2})$, each *sorting pass* is used to perform three actions: The first is to find all coefficients that are significant with respect to the current bitplane (these are coefficients that are larger in magnitude than the current threshold $\mathbf{2}^{\boldsymbol{n}}$ with $\boldsymbol{n} = \boldsymbol{n_{max}}, \boldsymbol{n_{max}} - \mathbf{1}, \boldsymbol{...}$). The second action is to transmit the significance-testing results and signs of those coefficients that have become significant against the current threshold. The final action is to update the initial spatial orientation trees by removing all significant coefficients and storing them separately. Here, the significance tests are performed on the basis of sets. If a set has become significant (at least one coefficient inside the set whose most significant bit locates at the current bitplane), a partitioning rule is used to partition the set into new subsets, then significance tests are performed on the new, smaller sets. This process continues until the significance test has been done for all significant sets, and the coordinates of all significant coefficients for the current bitplane have been identified. A succeeding *refinement pass* is used to transmit the current bitplane values for coefficients that are known to be significant from the previous bitplanes. The whole sorting and refinement-pass sequence is repeated until the desired bitrate is achieved or, in the case of lossless compression of finite-alphabet data, until all bitplanes have been transmitted.



(a) (b)

Figure 5.1: Parent-offspring dependencies in SPIHT. (a) 2-D tree for a seven-band wavelet transform. The larger squares represent the frequency bands ($\boldsymbol{LL_2, HL_2, LH_2, HH_2, HL_1, LH_1, HH_1}$) of the transformed image, and the smaller ones represent the individual coefficients within the bands. The arrows show the links between parents and their offspring. (b) 1-D tree following the offspring rule $\boldsymbol{\mathcal{O}(i) = iN + [0, N-1]}$ with $\boldsymbol{N = 4}$.

5.3.2 SPIHT-style Algorithm in Audio Compression

The idea of applying SPIHT-type significance trees to audio compression has been independently proposed in [107] and [34]. Both focused on compressing MDCT (modified discrete cosine transform) transformed audio signals and applied a parent-offspring relationship for the coefficient coordinates of the form $\boldsymbol{\mathcal{O}(i) = iN+[0, N-1]}$ where $\boldsymbol{i} \in \mathbb{N}^+$ is the parent coordinate and $\boldsymbol{N} \in \mathbb{N}^+$ is the number of offspring. This tree choice is somehow related to the fact that most instruments produce harmonics of a fundamental frequency, so that correlations might exist between the coefficients and their harmonics. Because this type of significance tree was inspired by the SPIHT algorithm, we will refer to it as the SPIHT-style significance tree in the following. Fig. 5.1(b) illustrates the SPIHT trees for $\boldsymbol{N = 4}$. In addition to SPIHT-related compression, additional perceptual significance tests were introduced in [108]. In this method, coefficients that were significant with respect to their magnitudes were only transmitted if they were also significant with respect to a masking threshold.

5.4 Description of the Dynamic Scalable Compression Scheme DSTQ

In this section, we consider the problem of constructing optimal significance trees for a 1-D transform vector and how to use them for compression. First we will explain the proposed DSTQ algorithm. Then we will present an approach to generate a data-dependent set of optimal significance trees.

5.4.1 DSTQ Algorithm

Let the vector $\mathbf{X} = (\boldsymbol{X_1}, \boldsymbol{X_2}, \cdots, \boldsymbol{X_M}), \boldsymbol{M} \in \mathbb{N}^+$ be the 1-D coefficient vector to be encoded, with the corresponding set of coefficient coordinates $\boldsymbol{\mathcal{M}} = [\mathbf{1}, \boldsymbol{M}]$. Here the data in the coefficient vector $\mathbf{X}$ is not specified. It can, for example, be real-valued signal samples, transform coefficients or integer-valued quantizer indices in a frame of audio. Similar to the SPIHT algorithm, our DSTQ algorithm encodes the coefficients subsequently bitplane for bitplane, commonly starting from their most significant and continuing to their least significant bitplane. The most significant bitplane $\boldsymbol{n_{MSB}}$ is determined by $\mathbf{2}^{\boldsymbol{n_{MSB}}} > \max_{\boldsymbol{i} \in \boldsymbol{\mathcal{M}}}(|\boldsymbol{X}_i|/\mathbf{2})$. DSTQ also distinguishes between a *sorting pass* (to select significant coefficients by tree-based significance mapping and output 'position' bits) and a *refinement pass* (to output 'value' bits).

Now let us assume that the coefficient-position information $\boldsymbol{\mathcal{M}}$ is mapped to a significance tree $\mathfrak{T}$. Then the so-called *sorting pass* performs the following significance-

tree tests for the current processed bitplane $\boldsymbol{n}$:

$$\boldsymbol{S(n)} = \begin{cases} \mathbf{1} & \text{if } \max\limits_{i \in \mathcal{M}}(|\boldsymbol{X}_i|) \geq \mathbf{2}^{\boldsymbol{n}} \\ \mathbf{0} & \text{otherwise} \end{cases} \tag{5.1}$$

The significance-tree test $\boldsymbol{S(n)}$ can be performed[2] calling the following pseudocode *TreeSignificance*($\mathfrak{T}$, $\mathbf{2}^{\boldsymbol{n}}$):

Algorithm 1 TreeSignificance(tree $\mathfrak{T}$, threshold $\boldsymbol{\tau}$)

```
1: if 𝔗 is not a leaf node then
2:    if 𝔗 is insignificant with respect to τ then
3:       emit '0' and return
4:    else
5:       emit '1'
6:    end if
7: end if
8: 𝔗^r ⇐ root node of 𝔗
9: if 𝔗^r is significant with respect to τ then
10:   emit '1' and sign bit
11: else
12:   emit '0'
13: end if
14: 𝔗_k^c ⇐ k-th child subtree of 𝔗^r
15: for all 𝔗_k^c do
16:   Call TreeSignificance(𝔗_k^c, τ)
17: end for
```

See also Figure 5.2 for a flow-chart representation of Algorithm 1. The important feature of this kind of coefficients-position mapping technique can be seen in the pseudocode of Algorithm 1 in the lines 2 and 3. We can save bit costs whenever we encode an insignificant tree (i.e., a tree with $\boldsymbol{S(n)} = \mathbf{0}$) with only one bit '0' instead of coding all the insignificant coefficients of the tree one-by-one. Even though there is

[2]The partitioning rule is slightly simplified compared to the 2-D one in SPIHT [111]

additional cost for transmitting the significance-tree test result for significant trees, due to the savings that occur when a tree is insignificant the method is, in general, more efficient than the straightforward one-by-one coding.

The proposed DSTQ algorithm compresses the coefficient set $\mathbf{X}$ in the order of threshold-by-threshold. At each bitplane, in the sorting pass, the procedure *TreeSignificance* is applied based on the dynamically selected local significance tree $\mathfrak{T}$. The coefficient positions that become significant in the current bitplane are determined and moved into a respective list of significant coefficients (LSC). In the refinement pass, also sequentially, we output the current bitplane values of those coefficients that became significant in the previous bitplanes. Then we move to the next lower bitplane, and the sequence of sorting and refinement passes is repeated. In this way, the DSTQ algorithm achieves encodings with finer and finer quantization steps by progressively transmitting the binary representation of the coefficients and yields a bit-wise scalable

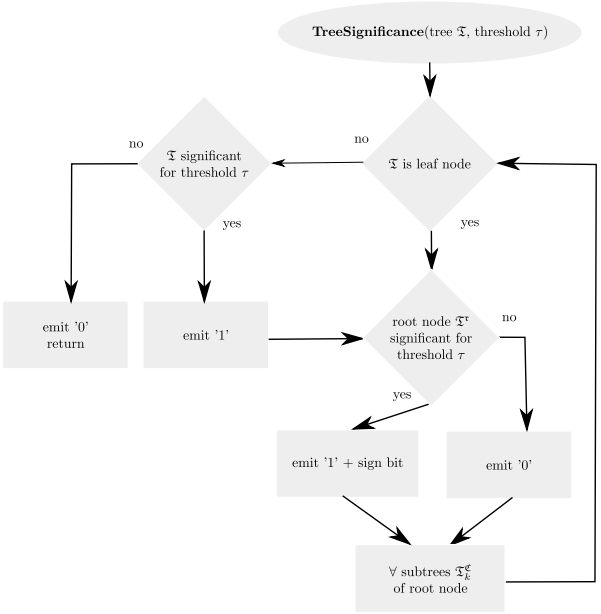


Figure 5.2: Flow-chart representation of Algorithm 1.

compression technique. The entire DSTQ algorithm is described as Algorithm 2 (See Figure 5.3 for a flow-chart representation).

Algorithm 2 DSTQ(coefficients $\mathbf{X}$, coordinates $\boldsymbol{\mathcal{M}}$)

Dynamic Tree Generation:
select optimal tree $\boldsymbol{\mathfrak{T}}$ from set of possible significance trees
Initialization:
output $\boldsymbol{n} = \lfloor \mathbf{log_2}(\max_{i \in \mathcal{M}}\{\mid X_i \mid\}) \rfloor$;
output index of selected $\boldsymbol{\mathfrak{T}}$
set LSC as an empty list.
while $\boldsymbol{n} \geq \mathbf{0}$ and bit budget is not fully utilized **do**
  Sorting Pass:
  call *TreeSignificance**($\boldsymbol{\mathfrak{T}}$,$\mathbf{2^n}$)
  store all newly significant coefficients into the LSC
  Refinement Pass:
  for all coefficient in LSCs except those included in the last sorting pass **do**
    output the $\boldsymbol{n^{th}}$ bit
  end for
  $\boldsymbol{n = n - 1}$
end while

The previous definition of *TreeSignificance* described a simplified version for comprehensibleness. In the *DSTQ* algorithm a modified version *TreeSignificance** is used that only emits test results for nodes that have not become significant in a previous bitplane. These coefficients are encoded in the refinement pass and their removal from the set of tested coefficient results in larger insignificant trees. The process is repeated until the desired rate (bit budget for compressing the coefficient set) is achieved, or, in case of lossless compression, all bits in all coefficients have been encoded. Like with the technique used in [111], to obtain the desired decoder's algorithm that duplicates the encoder's execution path, we simply have to replace the words 'output' by 'input' in the pseudo code.

The amount of side information required to transmit which significance tree has been selected is relatively low compared with the number of saved bits due the optimized significance tree. This will be confirmed by experimental results in Section 5.5, where we compare the performance of single fixed trees with the one of dynamically selected trees.

5.4.2 Data-driven Generation of Significance Trees

As stated before, in wavelet-based still image coding, due to the properties of natural images, the significance information can be well captured within a single type of significance tree. For audio, however, where the frequency content can change dramatically over time, there exists no single tree that captures the significance information of all

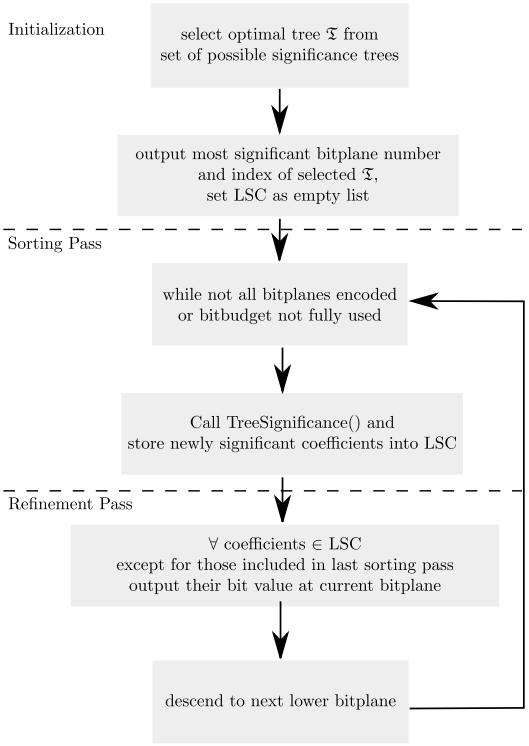


Figure 5.3: Flow-chart representation of Algorithm 2.

frames equally well. We therefore propose to dynamically select the best significance tree for each frame from a given set of possible trees. To efficiently construct such a set of significance trees, we suggest to directly use the information available from the coefficients distribution. We will first recapitulate some theoretical results on optimal significance trees for a given coefficients distribution.

For a significance tree $\mathfrak{T}$ the root node is denoted as $\mathfrak{T}^{\mathfrak{r}}$ and the k-th child subtree of $\mathfrak{T}^{\mathfrak{r}}$ is represented by $\mathfrak{T}_k^{\mathfrak{c}}$. The probability of significance of a tree with respect to the bitplane n is denoted as $P(\mathfrak{T}, n)$. The probability of significance of the root node at a given bitplane is written as $p(\mathfrak{T}^{\mathfrak{r}}, n)$. Then the average significance mapping cost of a tree $\mathfrak{T}$ for a bitplane n is given by

$$C(\mathfrak{T}, n) := 1 + P(\mathfrak{T}, n) \left(1 + \sum_k C(\mathfrak{T}_k^{\mathfrak{c}}, n) \right) \tag{5.2}$$

with

$$P(\mathfrak{T}, n) = 1 - (1 - p(\mathfrak{T}^{\mathfrak{r}}, n)) \prod_k \left(1 - P(\mathfrak{T}_k^{\mathfrak{c}}, n)\right) . \tag{5.3}$$

From (5.2) it follows that the encoding cost for a significance tree depends recursively on the probability of significance (5.3) of its child subtrees and the number of these recursive steps executed. This leads to the important conclusion that the optimal sequence of coefficient positions is in descending order of their magnitude, resulting in the largest possible insignificant subtrees for each bitplane.

We will illustrate this using the toy example given in Table 5.1. The sorting tree for this example, shown in Fig. 5.4, is mapping the coefficient positions depth-first in the order $\mathcal{M} = \{X_2, X_4, X_1, X_5, X_6, X_3\}$. An optimal tree would transmit for a bitplane all significant coefficients first, followed by the remaining insignificant coefficients. For the example, its sorting tree maps in the 3rd bitplane the significant

Coefficient		**Bitplane**		
Position	**Value**	3rd	2nd	1th
$\boldsymbol{X_1}$	2	0	1	0
$\boldsymbol{X_2}$	4	1	0	0
$\boldsymbol{X_3}$	0	0	0	0
$\boldsymbol{X_4}$	3	0	1	1
$\boldsymbol{X_5}$	1	0	0	1
$\boldsymbol{X_6}$	0	0	0	0

Table 5.1: Binary representation of the 1-D example signal.

coefficient $\boldsymbol{X_2}$ first and *TreeSignificance* will emit $\boldsymbol{\{11s\}}$ to the bitstream with $\boldsymbol{s}$ being the sign bit. The remaining coefficients of the bitplane are fully described by the test results for the subtrees with the root nodes $\boldsymbol{X_4}$ and $\boldsymbol{X_6}$. The bitstream encoding the MSB of the example is therefore $\boldsymbol{\{11s00\}}$. As for every bitplane its significant coefficient positions are moved to a list of significant coefficients (LSC) they are excluded from the sorting pass in the following bitplanes. Thereby all significant coefficients in the next bitplane will be of lower magnitude than the coefficients that became significant in the current bitplane. The sorting tree will map these coefficient first, in the example $\boldsymbol{X_2}$ is skipped in the next sorting pass and the subtree with the root node $\boldsymbol{X_4}$ is tested first, adding $\boldsymbol{\{11s\}}$ to the bitstream. Then the leaves mapping $\boldsymbol{X_1}$ and $\boldsymbol{X_5}$ are encoded and the insignificant coefficients $\boldsymbol{X_6}$ and $\boldsymbol{X_3}$ are encoded with a single $\mathbf{0}$ bit. In the refinement pass the next bit representing $\boldsymbol{X_2}$ is transmitted, resulting in the overall bitstream $\boldsymbol{\{11s0011s1s000\}}$. In the last bitplane, the sorting pass emits the test result for the subtree with the root node $\boldsymbol{X_6}$. With the final refinement pass the bitstream of the example is $\boldsymbol{\{11s0011s1s00000101\}}$.

It can be concluded that an optimal significance tree for a given 1-D coefficient vector can be derived by computing its sorting tree. In audio coding applications the

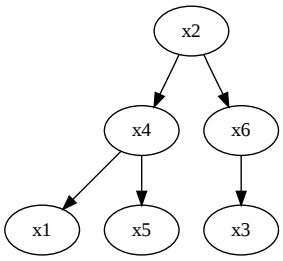


Figure 5.4: Significance tree of the 1-D example which is its sorting tree.

signal is commonly divided into frames of fixed length. To derive a set of optimal significance trees for such frames, we define a training dataset, and the sorting trees for every frame of the training dataset are computed. If the DSTQ coding scheme is applied for an entire signal class, for example for human speech in a telecommunication scenario, the training dataset would be a speech database. If the signal to be encoded is known *a priori*, like it is the case for storing digital audio, the signal itself can be used as the training dataset. We propose the following simple algorithm to learn a set of significance trees that is optimized for the training dataset. First we encode the $\boldsymbol{F}$ frames of the training dataset with all available $\boldsymbol{F}$ sorting trees derived from these frames. The performance of every tree is measured using the lengths of the resulting bitstreams needed to achieve lossless encoding. These derived metric values are stored into a $\boldsymbol{F} \times \boldsymbol{F}$ matrix $\mathbf{M}$. It is to note that the DSTQ algorithm has a low complexity – a simple C prototype implementation on a standard workstation was able to test 10000 trees per second for a frame length of 1024 samples. We then use the following algorithm to reduce the number of significance trees to $\boldsymbol{K}$ trees:

Note that instead of the length of the bitstream achieving lossless compression a perceptual metric could be used. And the computational complexity could be reduced

Algorithm 3 Derive set of $\boldsymbol{K}$ optimal significance trees

1: find in $\mathbf{M}$ for every frame the tree achieving the best performance and store their ID in $\mathbf{T} \in \mathbb{N}^{\boldsymbol{F}}$
2: **while** (number of unique trees in $\mathbf{T}$) $> \boldsymbol{K}$ **do**
3: get for every tree in $\mathbf{T}$ its next best tree for the according frame from $\mathbf{M}$ and store its ID in $\mathbf{T}' \in \mathbb{N}^{\boldsymbol{F}}$
4: replace in $\mathbf{T}$ the tree whose next best tree in $\mathbf{T}'$ is already used in $\mathbf{T}$ and which results in the smallest performance degeneration for its frame.
5: if no such tree is found in $\mathbf{T}'$ search for the second next best / third next best / ... tree.
6: **end while**

by limiting the iterations of the DSTQ scheme to the first $\boldsymbol{n}$ bitplanes or to a given bitrate. If both the encoder and the decoder know the $\boldsymbol{F}$ previously decoded frames to a certain minimal reconstruction level, it is possible to use this information to derive new optimal significance trees after each audio frame which can replace trees in the set that have seldom been selected for the past frames. Another alternative to dynamically update the set of significance trees is to use a side-information channel that continuously transmits updates to the set of significance trees.

5.5 Experimental Results

We applied our proposed DSTQ scheme to the compression of audio signals, and in this section, we compare our method with the existing algorithms SPIHT [107, 34] and CSTQ [127] for lossy audio compression. In addition, we combined our method with the state-of-the-art MPEG-2/4 AAC compression scheme by replacing the Huffman coding stage for quantizer-index compression by the SPIHT, CSTQ and DSTQ algorithms. Performance comparisons for this AAC-related scheme are made with the standardized MPEG-2/4 AAC scheme with low-complexity profile (fixed bitrate) and the scalable

MPEG-4 BSAC and MPEG-4 SLS schemes.

5.5.1 Comparison With Schemes Using a priori Fixed Trees

In this experiment, we compared the compression performance of our proposed DSTQ algorithm with the single-tree SPIHT algorithm and the CSTQ coding scheme which uses a set of *a priori* fixed trees [127]. The CSTQ algorithm separates an audio frame into eight segments and selects for each segment a tree that assumes a descending, ascending, convex or concave coefficient-magnitude behavior. This results in a set of 65536 possible trees for every frame. The a capella song "Tom's Diner" by Suzanne Vega was transformed with the MDCT filterbank with $\boldsymbol{M = 1024}$ frequency bands and then encoded with the SPIHT, CSTQ and DSTQ algorithms, respectively. All coding schemes were performed with significance trees having a tree order of four. For the DSTQ algorithm, a full search over all trees was performed for every audio frame, selecting the tree that results in the shortest bitstream for a lossless encoding. Sets of optimal significance trees consisting of 32, 64, 128, 256, and 512 trees have been derived as described in Algorithm 3. To allow for a direct comparison of the significance trees, the same tree-selection algorithm was used for CSTQ in contrast to [127], where a less complex tree selection algorithm was applied. The test bitrates were varied between 16 and 96 kbps. The according frame bit budget $\boldsymbol{R_f}$ was computed as $\boldsymbol{R_f = \lfloor R \cdot M / Fs \rfloor}$ where $\boldsymbol{R}$ is the required bitrate in bits per second, and $\boldsymbol{Fs}$ is the sampling rate in Hz. As a quality measure for the compression performance, the

segmental signal-to-noise ratio (segSNR) was used, which was computed as follows:

$$\boldsymbol{segSNR} = \frac{\mathbf{1}}{\boldsymbol{N}} \sum_{\boldsymbol{j=0}}^{\boldsymbol{N-1}} \mathbf{10}\log_{\mathbf{10}} \frac{\sum_{\boldsymbol{i=1}}^{\boldsymbol{M}} \left(\boldsymbol{X}_i^{(j)}\right)^{\mathbf{2}}}{\sum_{\boldsymbol{i=1}}^{\boldsymbol{M}} \left(\boldsymbol{X}_i^{(j)} - \hat{\boldsymbol{X}}_i^{(j)}\right)^{\mathbf{2}}} \tag{5.4}$$

where $\boldsymbol{X}_i^{(j)}$, $\boldsymbol{i} \in [\mathbf{1}, \boldsymbol{M}]$ are the transform coefficients in frame $\boldsymbol{j}$ and $\hat{\boldsymbol{X}}_i^{(j)}$ are the corresponding reconstructed coefficients. $\boldsymbol{N}$ is the number of frames. A psychoacoustic analysis was not carried out for this experiment, because all schemes applied the same MDCT and significance tests (i.e., the same thresholds), so that, in this special case, the segmental SNR indeed allows for a comparison between the different significance tree related compression techniques.

The results presented in Table 5.2 show that algorithms using a set of possible significance trees achieve a better segmental SNR than single-tree algorithms like SPIHT. The number of saved bits due to the selection of significance-trees being optimal for the current frame is larger than the amount of side information needed to transmit which significance-tree has been selected. It further shows that using a data-driven approach to construct a set of trees being optimal for a specific class of signals (DSTQ), a higher compression performance can be achieved with a smaller set of possible significance trees compared to a model-driven approach (CSTQ). For bitrates equal to or less than 32 kbps a set of 256 optimized significance trees achieved a similar performance as the set of 65536 model-driven trees (CSTQ). For higher bitrates, a set of 512 learned significance trees showed a superior signal reconstruction performance than CSTQ. For digital audio storage applications, the derived optimal set can be placed at the beginning of the media and only the ID of the selected tree needs to be transmitted for every audio frame. Using a simple run-length encoding

Algorithm	tree ID coding cost	bitrate					
		16kbps	32kbps	48kbps	64kbps	80kbps	96kbps
SPIHT (1 tree)	0 bit	15.121	19.268	22.817	26.073	29.081	31.945
CSTQ (65536 trees)	16 bit	16.116	20.641	24.352	27.710	30.846	33.873
DSTQ (512 trees)	9 bit	17.865	22.577	26.329	29.638	32.627	35.348
DSTQ (256 trees)	8 bit	16.536	20.607	23.939	26.963	29.787	32.419
DSTQ (128 trees)	7 bit	15.786	19.602	22.771	25.680	28.427	31.018
DSTQ (64 trees)	6 bit	15.345	19.025	22.132	24.990	27.698	30.266
DSTQ (32 trees)	5 bit	15.073	18.687	21.759	24.602	27.295	29.865

Table 5.2: Segmental SNRs in dB for the svega test signal encoded at bitrates between 16-96 kbps using different algorithms.

we needed at most 5000 bits to encode a significance tree for a frame length of 1024 samples. This would result in a maximal cost of 2.44 MB for 512 trees which is below 0.5% of the capacity of a standard audio CD. And for applications like speech coding it is not necessary to transmit the learned set of significance trees, as an *a priori* learned set of significance trees can be stored in the decoding device. Also a constant update of the significance tree set can be performed as proposed in Subsection 5.4.2.

5.5.2 Comparison with MPEG-2/4 AAC, MPEG-4 BSAC and MPEG-4 SLS

In this section, we compare the perceptual quality of our proposed DSTQ scheme with the single-tree SPIHT algorithm, the CSTQ coding scheme, the standardized MPEG-2/4 AAC fixed-bitrate encoder and the MPEG-4 BSAC and MPEG-4 SLS scalable encoders, respectively. Within the MPEG-2/4 AAC compression scheme, a flexible Huffman codebook selection (from 11 pre-designed Huffman codebooks) is adopted to losslessly compress the quantizer indices. The MPEG-4 BSAC coder uses an alternative noiseless coding method (bit slice arithmetic coding instead of Huffman coding), with

the rest of the processing (filterbank, psychoacoustic model, etc.) being identical to MPEG-2/4 AAC. The MPEG-4 BSAC coder is designed to support scalability with nearly transparent sound quality at 64 kbps and graceful degradation at lower bitrates and performs best in the range of 40 kbps to 64 kbps. The MPEG-4 SLS coder realizes a two-layer structure with a Huffman encoded AAC core and a lossless enhancement bitstream which is encoded using context-based bitplane arithmetic coding and a special entropy encoding mode for low-energy frames. MPEG-4 SLS is designed to achieve lossless coding at a bitrate of approximately 350 kbps/channel and nearly transparent sound quality at approximately 64kbps as a result of the AAC core layer. In our experiments, MPEG-2/4 AAC and MPEG-4 BSAC compression procedures were implemented based on the MPEG-2/4 reference software (2001 Edition) with available source codes on [60]. The MPEG-4 SLS coding scheme was implemented using the available source code on [61]. Due to the poor performance of the reference software we further included a state-of-the-art MPEG-2/4 AAC implementation (Nero AAC codec 1.3.3.0, http://www.nero.com) in our test setup.

For a fair comparison, we adopted the basic MPEG-2/4 AAC encoding process before noiseless coding for our DSTQ coder as well. That is, we kept the MPEG-2/4 AAC scheme unchanged up to the point where Huffman coding is employed, but instead of using Huffman codebooks, our proposed DSTQ algorithm was employed for quantizer-index compression. In detail, the MDCT coefficients that have been quantized according to the psychoacoustic model are not Huffman encoded but compressed with the DSTQ coding scheme. The resulting bitstream consists of the MPEG-2/4 sideband information, e.g. the chosen scalefactors and the DSTQ coefficients that take up the same space in the new bitstream as the Huffman coefficients did in the original MPEG-2/4 AAC bitstream. This combined scheme will be referred to as the AAC-DSTQ

scheme in the following. The CSTQ and SPIHT coding schemes have been realized in the same manner and will be referred to as AAC-CSTQ and AAC-SPIHT. To be able to compare the audio-coding performance of MPEG-4 SLS at low bit rates we chose for the AAC core layer a bitrate of 16 kbps. This allowed also a direct comparison with MPEG-4 BASC whose base layer is encoded at 16 kbps. It is to note that for MPEG-4 SLS higher AAC core layer bitrates have not been investigated in this work as the main focus is on coding scenarios with a scalability form low up to high bitrates which leads to the necessity of a low-bitrate AAC core in the MPEG-4 SLS scheme.

We evaluated the audio coding schemes using a set of five test files with different characteristics. From the SQAM test material [36] we selected a male German speaker (track #53), a recording of a harpsichord as an example for a strong harmonic sound structure (track #40), and a castanets recording (track #27) consisting of sharp attacks. We further selected the pop song "Mountains O'Things" by Tracy Chapman and the a capella song "Tom's Diner" by Suzanne Vega.

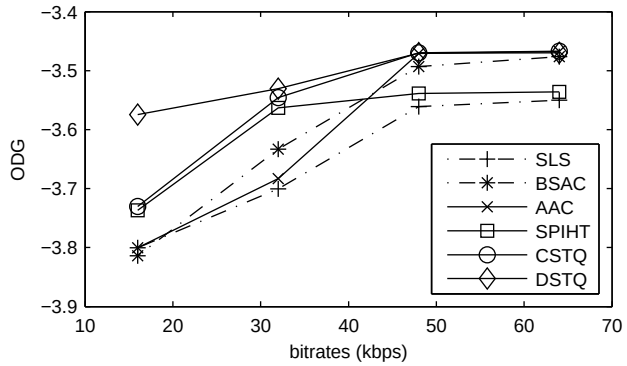


Figure 5.5: MPEG-2/4 reference software: Objective difference grades for the set of audio test files encoded with MPEG-4 SLS, MPEG-4 BSAC, MPEG-2/4 AAC and AAC-SPIHT, AAC-CSTQ and AAC-DSTQ using quantizer indices derived at 64kbps.

In the first experiment we used the MPEG-2/4 reference software. We applied AAC-DSTQ, AAC-CSTQ and AAC-SPIHT to encode the quantizer indices derived by the reference software at 64 kbps, matching the maximum bitrate of MPEG-4 BSAC, and then truncated the bitstream to the different target bitrates. In the MPEG-4 SLS coding scheme the AAC core bitrate was set to 16 kbps to allow for a minimal bitrate of 16 kbps. We measured the perceived audio quality of the decoded audio signals relative to the original test signal using a model of auditory perception (PEMO-Q) [55]. The estimated perceived audio quality was mapped to a single quality indicator, the Objective Difference Grade (ODG) [66]. This is a continuous scale from **0** for "imperceptible impairment", **−1** for "perceptible but not annoying impairment", **−2** for "slightly annoying impairment", **−3** for "annoying impairment" to **−4** for "very annoying impairment". For the AAC-DSTQ, AAC-CSTQ and AAC-SPIHT coders, the bitstreams in each frame consisted of the side information produced by the MPEG-2/4 AAC encoder when the total bit rate was selected to be 64 kbps, the tree-selection bits if applicable, and the embedded bitstream truncated to meet the target bitrates. It is to note that the encoding for AAC-DSTQ, AAC-CSTQ and AAC-SPIHT was performed thereby only once for the highest bitrate and that the decoding at the lower bitrates was performed by simply truncating this bitstream to the desired bitrate. For the MPEG-2/4 AAC coder the complete encoding and decoding process was repeated for all lower bitrates. The tree order for all significance-tree based coding schemes was set to four. All signals were encoded as mono signals and temporal noise shaping was not used.

From the resulting objective difference grades for the set of audio test files shown in Fig. 5.5 we can see that MPEG-2/4 AAC, MPEG-4 BSAC, AAC-CSTQ and AAC-DSTQ achieved a similar perceived audio quality from 64 to 48 kbps with MPEG-4 BSAC

showing a marginally lower perceived audio quality at 48 kbps. For AAC-SPIHT lower ODGs were derived from 64 to 48 kbps that showed a parallel trend to the ODGs of AAC-CSTQ and AAC-DSTQ. MPEG-4 SLS resulted in the lowest ODGs from 64 to 32 kbps. MPEG-2/4 AAC resulted in the second lowest perceived audio quality for 32 kbps, followed by MPEG-4 BSAC, AAC-SPIHT, AAC-CSTQ and AAC-DSTQ. For 16 kbps MPEG-4 BSAC resulted in the lowest ODG and a similar perceived audio quality was derived for MPEG-4 SLS and MPEG-2/4 AAC. AAC-DSTQ showed for all bitrates the best perceived audio quality. Interestingly the MPEG-2/4 AAC coder of the reference implementation achieved low ODGs for bitrates below 48 kbps despite

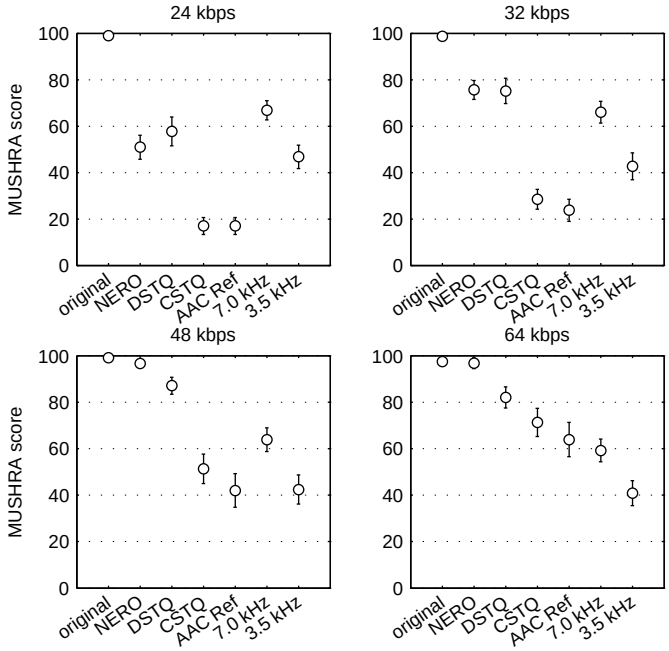


Figure 5.6: MPEG-2/4 AAC Nero implementation: Results of MUSHRA listening tests for the set of audio test files encoded with MPEG-2/4 AAC Nero, AAC-DSTQ and AAC-CSTQ using quantizer indices derived at 64kbps and the hidden reference and anchors. Error bars denote 95% confidence intervals for mean.

the fact that it was allowed to optimize the quantizer indices for every target bitrate separately in contrast to the other coding schemes. It showed that the rate loop of the MPEG-2/4 AAC reference implementation did not optimally utilize the available bit budget at lower bitrates.

Therefore we conducted a second experiment using the state-of-the-art Nero AAC encoder. We newly encoded the set of audio test files using MPEG-2/4 AAC Nero and analogous to the previous experiment these quantizer indices derived for 64 kbps were encoded using CSTQ and DSTQ. We further used a reduced number of audio coding schemes (MPEG-2/4 AAC, CSTQ-AAC and DSTQ-AAC) to allow the evaluation of the perceived audio quality with the resource-demanding subjective listening test method MUSHRA (multi stimuli with hidden reference and anchor points) [67]. The tests were performed using AKG K240 headphones, Creative Sound Blaster Audigy sound cards and the ABC/Hidden Reference Audio Comparison Tool [92]. The 12 listeners performed a training session of approximately 15 min and had the opportunity to adjust the playback level only within this training period. Test instructions explained the user interface of the software and how to give the scores on the quality scale from 1 (bad) to 100 (excellent). The chosen anchor points were low-pass-filtered originals with cutoff frequencies of 3.5 kHz and 7 kHz.

From the resulting MUSHRA scores for the set of audio test files shown in Fig. 5.6 we can see that for 64 kbps MPEG-2/4 AAC Nero achieved a transparent audio coding quality, followed by AAC-DSTQ, AAC-CSTQ and the reference implementation of MPEG-2/4 AAC. The same ranking order was derived for 48 kbps. For 32 kbps MPEG-2/4 AAC Nero and AAC-DSTQ achieved a similar MUSHRA score and for 24 kbps AAC-DSTQ achieved the best perceptual audio quality, followed by MPEG-2/4 AAC Nero and AAC-CSTQ and the reference implementation of MPEG-2/4 AAC

achieved the lowest MUSHRA scores.

Overall, the perceptual quality for AAC-DSTQ, especially at low bitrates, was better than for the competing scalable coding schemes and for bitrates below 32 kbps a higher perceptual coding quality than the non-scalable MPEG-2/4 AAC coding scheme was achieved. This indicates that the signal-dependent significance trees of the DSTQ scheme can reconstruct the important coefficients earlier than audio coding schemes assuming more general coefficient distributions.

5.6 Conclusions

The fine-grain scalable 1-D signal compression problem has been addressed in this study. While in almost all existing SPIHT-related algorithms a single-type significance tree has been adopted for sorting significant coefficients and transmitting position information, we have proposed a novel dynamic significance tree technique, which learns optimal tree structures for 1-D signal compression. Based on the selection of an optimal tree from a learned set of significance trees, a compression scheme called DSTQ has been developed providing high compression quality and bitrate scalability. Further, we applied our proposed scheme to audio signal compression. Here, the advantages of our proposed scheme are clearly demonstrated: the method outperforms the existing SPIHT-like algorithm and yields competitive results for compressing quantizer indices in the MPEG-2/4 AAC audio compression scheme, yet with fine-grain scalability.

Acknowledgment

The authors would like to thank the anonymous reviewers for their constructive comments and corrections and the MUSHRA listening test participants for their effort. The author Stefan Strahl wants to especially thank Astrid Klinge for her support on this manuscript and wants to ask her to marry him.

Bibliography

[1] S. Abdallah and M. Plumbley, "If edges are the independent components of natural images, what are the independent components of natural sounds," *International Conference On Independent Component Analysis and Blind Signal Separation*, 2001.

[2] A. M. H. J. Aertsen and P. I. M. Johannesma, "Spectro-temporal receptive fields of auditory neurons in the grassfrog," *Biological Cybernetics*, vol. 38, no. 4, pp. 223–234, Nov. 1980.

[3] M. Aharon, M. Elad, and A. Bruckstein, "K-SVD: An Algorithm for Designing of Overcomplete Dictionaries for Sparse Representation," *IEEE Transactions on Signal Processing*, vol. 54, no. 11, pp. 4311–4322, 2006.

[4] E. Ambikairajah, J. Epps, and L. Lin, "Wideband speech and audio coding using gammatone filter banks," in *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2001, pp. 773–776.

[5] Apple Inc., "Airtunes," 2004, data last viewed 23. April 2009. [Online]. Available: http://www.apple.com/airportexpress/features/airtunes.html

[6] J. Atick, "Could information theory provide an ecological theory of sensory processing?" *Network: Computation in neural systems*, vol. 3, no. 2, pp. 213–251, 1992.

[7] D. Attwell and S. Laughlin, "An energy budget for signaling in the grey matter of the brain," *J Cereb Blood Flow Metab*, vol. 21, no. 10, pp. 1133–45, 2001.

[8] H. Barlow, "Possible principles underlying the transformation of sensory messages," in *Sensory Communication*, W. A. Rosenbluth, Ed. MIT Press, Cambridge, 1961, pp. 217–234.

[9] A. Bell and T. Sejnowski, "Learning the higher order structure of a natural sound," *Network: Computation in Neural Systems*, vol. 7, no. 2, pp. 261–266, 1996.

[10] Bluetooth Audio Video Working Group, "Advanced audio distribution profile specification," May 2003, version 1.00.

[11] H. Bolcskei and F. Hlawatsch, "Oversampled filter banks: optimal noise shaping, design freedom, and noise analysis," in *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, vol. 3, 1997, pp. 2453–2456.

[12] H. Bolcskei, F. Hlawatsch, and H. Feichtinger, "Frame-theoretic analysis of oversampled filter banks," *IEEE Transactions on Signal Processing*, vol. 46, no. 12, pp. 3256–3268, 1998.

[13] K. Brandenburg, O. Kunz, and A. Sugiyama, "MPEG-4 natural audio coding," *Signal Processing: Image Communication*, vol. 15, no. 2, pp. 423–444, 2000.

[14] M. Brucke, W. Nebel, A. Schwarz, B. Mertsching, M. Hansen, and B. Kollmeier, "Silicon cochlea: A digital VLSI implementation of a quantitative model of the auditory system," *J. Acoust. Soc. Am*, vol. 105, p. 1192, 1999.

[15] T. C. T. Carney, L. H. & Yin, "Temporal coding of resonances by low-frequency auditory nerve fibers: single-fiber responses and a population model," *J. Neurophys.*, vol. 60, p. 16531677, 1988.

[16] S. S. Chen, D. L. Donoho, and M. A. Saunders, "Atomic decomposition by basis pursuit," *SIAM Journal on Scientific Computing*, vol. 20, no. 1, pp. 33–61, 1998.

[17] S. S. Chen, "Basis Pursuit," Ph.D. dissertation, Stanford University, 1995.

[18] T. Chi, P. Ru, and S. Shamma, "Multiresolution spectrotemporal analysis of complex sounds," *J. Acoust. Soc. Am*, vol. 118, no. 2, pp. 887–906, 2005.

[19] M. Cooke, "A glimpsing model of speech perception in noise," *J. Acoust. Soc. Am*, vol. 119, p. 1562, 2006.

[20] R. Crochiere and L. Rabiner, *Multirate digital signal processing.* Prentice-Hall Englewood Cliffs, NJ, 1983.

[21] Z. Cvetkovic and M. Vetterli, "Overcomplete expansions and robustness," in *Proceedings of the IEEE International Symposium on Time-Frequency and Time-Scale Analysis*, 18–21 June 1996, pp. 325–328.

[22] T. Dau, D. Püschel, and A. Kohlrausch, "A quantitative model of the effective signal processing in the auditory system. I. Model structure," *J. Acoust. Soc. Am*, vol. 99, no. 6, pp. 3615–3622, 1996.

[23] ——, "A quantitative model of the effective signal processing in the auditory system. II. Simulations and measurements," *J. Acoust. Soc. Am*, vol. 99, no. 6, pp. 3623–3631, 1996.

[24] I. Daubechies, "The wavelet transform, time-frequency localization and signal analysis," *IEEE Transactions on Information Theory*, vol. 36, no. 5, pp. 961–1005, 1990.

[25] ——, *Ten Lectures on Wavelets*. Philadelphia: SIAM: Society for Industrial and Applied Mathematics, 1992.

[26] I. Daubechies, A. Grossmann, and Y. Meyer, "Painless nonorthogonal expansions," *Journal of Mathematical Physics*, vol. 27, pp. 1271–1283, 1986.

[27] M. Davies and L. Daudet, "Sparse audio representations using the MCLT," *Signal Processing*, vol. 86, no. 3, pp. 457–470, 2006.

[28] G. Davis, "Adaptive Nonlinear Approximations," Ph.D. dissertation, New York University, 1994.

[29] E. de Boer, "On the principle of specific coding," *J Dyn Syst Meas Control Trans ASME*, vol. 95, pp. 265–273, 1973.

[30] I. Djokovic and P. Vaidyanathan, "Results on biorthogonal filter banks," *Applied and computational harmonic analysis*, vol. 1, pp. 329–343, 1994.

[31] D. Donoho, M. Elad, and V. Temlyakov, "Stable recovery of sparse overcomplete representations in the presence of noise," *Information Theory, IEEE Transactions on*, vol. 52, no. 1, pp. 6–18, 2006.

[32] D. Donoho and Y. Tsaig, "Recent advances in sparsity-driven signal recovery," in *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP05)*, 2005.

[33] R. Duffin and A. Schaeffer, "A class of nonharmonic Fourier series," *Trans. Amer. Math. Soc*, vol. 72, no. 2, pp. 341–366, 1952.

[34] C. Dunn, "Efficient audio coding with fine-grain scalability," in *Proc. AES 111th Convention*. New York, USA: preprint 5492, September 2001.

[35] B. Edler and G. Schuller, "Audio coding using a psychoacoustic pre-and post-filter," in *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, vol. 2, 2000, pp. 881–884.

[36] European Broadcasting Union, "Sound quality assessment material (SQAM) recordings for subjective tests," Tech. Rep. 3253, 1988. [Online]. Available: http://sound.media.mit.edu/mpeg4/audio/sqam/

[37] C. Feldbauer, G. Kubin, and W. Kleijn, "Anthropomorphic Coding of Speech and Audio: A Model Inversion Approach," *EURASIP Journal on Applied Signal Processing*, vol. 9, pp. 1334–1349, 2005.

[38] S. E. Ferrando, L. A. Kolasa, and N. Kovačević, "Algorithm 820: A flexible implementation of matching pursuit for gabor functions on the interval," *ACM Trans. Math. Softw.*, vol. 28, no. 3, pp. 337–353, 2002.

[39] J. Flanagan, "Models for Approximating Basilar Membrane Displacement," *J. Acoust. Soc. Am*, vol. 32, p. 937, 1960.

[40] J. B. J. Fourier, *Théorie analytique de la chaleur (The Analytical Theory of Heat)*. Paris: F. Didot, 1822.

[41] P. Frossard, P. Vandergheynst, R. Figueras i Ventura, and M. Kunt, "A posteriori quantization of progressive matching pursuit streams," *IEEE Transactions on Signal Processing*, vol. 52, no. 2, pp. 525–535, 2004.

[42] D. Gabor, "Theory of communications," *Journal of Institute of Electrical Engineers*, vol. 93, no. III-26, pp. 429–457, 1946.

[43] J. Garofolo, L. Lamel, W. Fisher, J. Fiscus, D. Pallet, and N. Dahlgren, "The DARPA TIMIT acoustic-phonetic continuous speech corpus CDROM. NTIS order number PB91-505065," 1990.

[44] M. M. Goodwin, *Adaptive Signal Models: Theory, Algorithms, and Audio Applications*. Boston: Kluwer Academic Publishers, 1998.

[45] I. Gorodnitsky and B. Rao, "Sparse signal reconstruction from limited data using FOCUSS: are-weighted minimum norm algorithm," *IEEE Transactions on Signal Processing*, vol. 45, no. 3, pp. 600–616, 1997.

[46] V. Goyal, M. Vetterli, and N. Thao, "Quantized overcomplete expansions in IR N: analysis, synthesis, and algorithms," *IEEE Transactions on Information Theory*, vol. 44, no. 1, pp. 16–31, 1998.

[47] D. Greenwood, "Critical Bandwidth and the Frequency Coordinates of the Basilar Membrane," *The Journal of the Acoustical Society of America*, vol. 33, p. 1344, 1961.

[48] R. Gribonval, "Approximations non-linéaires pour l'analyse des signaux sonores," Ph.D. dissertation, Université Paris IX Dauphine, 1999.

[49] ——, "Fast matching pursuit with a multiscale dictionary of Gaussian chirps," *IEEE Transactions on Signal Processing*, vol. 49, no. 5, pp. 994–1001, May 2001.

[50] R. Gribonval and S. Krstulovic, "MPTK, The Matching Pursuit Toolkit," 2005, data last viewed 23. April 2009. [Online]. Available: http://mptk.gforge.inria.fr/

[51] T. Herzke and V. Hohmann, "Improved Numerical Methods for Gammatone Filterbank Analysis and Synthesis," *Acta Acustica united with Acustica*, vol. 93, no. 3, pp. 498–500, 2007.

[52] P. Hoang and P. Vaidyanathan, "Non-uniform multirate filter banks: theory and design," in *Proceedings of the IEEE International Symposium on Circuits and Systems*, 1989, pp. 371–374.

[53] V. Hohmann, "Frequency analysis and synthesis using a Gammatone filterbank," *Acta Acustica united with Acustica*, vol. 88, no. 3, pp. 433–442, 2002.

[54] P. Hoyer, "Non-negative sparse coding," *Proceedings of the 12th IEEE Workshop on Neural Networks for Signal Processing*, pp. 557–565, 2002.

[55] R. Huber and B. Kollmeier, "PEMO-Q: A new method for objective audio quality assessment using a model of auditory perception," *IEEE Transactions on Audio, Speech and Language Processing*, vol. 14, no. 6, pp. 1902–1911, 2006.

[56] T. Irino and R. Patterson, "A dynamic, compressive gammachirp auditory filterbank," *IEEE Trans. on Audio, Speech, and Language Processing*, vol. 14, no. 6, pp. 2222–2232, 2006.

[57] T. Irino and M. Unoki, "A time-varying, analysis/synthesis auditory filterbank using the gammachirp," in *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, vol. 6, 1998, pp. 3653–3656.

[58] T. Irino, "An optimal auditory filter," in *Proceedings of the IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA)*, 1995, pp. 198–201.

[59] ISO, *ISO 389-7 Acoustics-Reference zero for the calibration of audiometric equipment-Part 7: Reference threshold of hearing under free-field and diffuse-field listening conditions*, International Organization for Standardization, Geneva, 1996.

[60] ISO/MPEG, "ISO/IEC 14496-5:2001 - Information technology – Coding of audio-visual objects – Part 5: Reference software," data last viewed 13. January 2010. [Online]. Available: http://www.iso.ch/iso/en/ittf/PubliclyAvailableStandards/ISO_IEC_14496-5_2001_Software_Reference/

[61] ——, "ISO/IEC 14496-5:2001/Amd.10:2007 - Information technology – Coding of audio-visual objects – Part 5: Reference software," data last viewed 13. January 2010. [Online]. Available: http://standards.iso.org/ittf/PubliclyAvailableStandards/c043465_ISO_IEC_14496-5_2001_Amd_10_2007_Reference_Software.zip

[62] ——, "Audio test report. ISO/IEC/JTC 1/SC 2/WG 11 MPEG MPEG90/N0030," 1990.

[63] ——, "MPEG-4 Audio Version 2 ISO/IEC 14496-3:1999/Amd.1," 1999.

[64] ——, "Scalable Lossless Coding (SLS) ISO/IEC 14496-3:2005/Amd.3:2006," 2006. [Online]. Available: http://standards.iso.org/ittf/PubliclyAvailableStandards/c043465_ISO_IEC_14496-5_2001_Amd_10_2007_Reference_Software.zip

[65] ITU-R Recommendation BS.1116-1, "Methods for the subjective assessment of small impairments in audio systems including multichannel sound systems," *ITU Publication*, Dec. 1997.

[66] ITU-R Recommendation BS.1387-1, "Methods for Objective Measurements of Perceived Audio Quality," *ITU Publication*, 2001.

[67] ITU-R Recommendation BS.1534, "Method for the subjective assessment of intermediate quality level coding systems," *ITU Publication*, 2001.

[68] ITU-T Recommendation G.722, "7khz audio coding within 64 kbit/s using Sub-band Adaptive Differential Pulse Code Modulation (SB-ADPCM)," *ITU Publication*, 1988.

[69] ITU-T Recommendation G.727, "5-, 4-, 3- and 2-bit/sample embedded adaptive differential pulse code modulation (ADPCM)," *ITU Publication*, 1990.

[70] P. I. Johannesma, "The pre-response stimulus ensemble of neurons in the cochlear nucleus," in *Symposium on Hearing Theory*. Eindhoven, Holland: Institute for Perception Research (IPO), June 1972, pp. 58–69.

[71] B. Kovesi, D. Massaloux, and A. Sollaud, "A scalable speech and audio coding scheme with continuous bitrate flexibility," in *Proc. Int. Conf. on Acoustics, Speech, and Signal Processing (ICASSP'04)*, vol. 1, Montreal, Canada, May 2004, pp. I–273–6.

[72] S. Krstulovic and R. Gribonval, "MPTK: Matching Pursuit Made Tractable," *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP06)*, 2006.

[73] G. Kubin and W. Kleijn, "Multiple-description coding (MDC) of speech with an invertibleauditory model," in *Proceedings of the IEEE Workshop on Speech Coding*, 1999, pp. 81–83.

[74] ——, "On speech coding in a perceptual domain," in *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, 1999, pp. 205–208.

[75] S. B. Laughlin and T. J. Sejnowski, "Communication in neuronal networks." *Science*, vol. 301, no. 5641, pp. 1870–1874, September 2003.

[76] M. Lewicki, "Efficient coding of natural sounds," *Nature Neuroscience*, vol. 5, no. 4, pp. 356–363, 2002.

[77] A. S. Lewis and G. Knowles, "A 64 kb/s video codes using the 2-D wavelet transform," in *Proc. Data Compression Conf.*, Snowbird, Utah, USA, 1991, pp. 196–201.

[78] J. Li, "Embedded audio coding (EAC) with implicit auditory masking," in *Proc. ACM on Multimedia*, Nice, France, December 2002, pp. 592–601.

[79] T. Li, S. Rahardja, and S. N. Koh, "Frequency region-based prioritized bit-plane coding for scalable audio," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 16, no. 1, pp. 94–105, 2008.

[80] X. Li and X. Zhuang, "The decay and correlation properties in wavelet transform," *Tech. Rep., Univ. Missouri-Columbia, USA*, March 1997.

[81] L. Lin, W. Holmes, and E. Ambikairajah, "Auditory filter bank inversion," in *Proceedings of the IEEE International Symposium on Circuits and Systems (ISCAS)*, vol. 2, 2001, pp. 537–540.

[82] Z. Liu and L. J. Karam, "Quantifying the intra and inter subband correlations in the zerotree-based wavelet image coders," in *Conf. Record of the 36th Asilomar Conf. on Signals, Systems and Computers*, Pacific Grove, USA, September 2002, pp. 1730–1734.

[83] C. Loeffler and C. Burrus, "Optimal design of periodically time-varying and multirate digital filters," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 32, no. 5, pp. 991–997, 1984.

[84] E. Lopez-Poveda and R. Meddis, "A human nonlinear cochlear filterbank," *J. Acoust. Soc. Am.*, vol. 110, pp. 3107–3118, 2001.

[85] Z. Lu and W. A. Pearlman, "An efficient, low-complexity audio coder delivering multiple levels of quality for interactive applications," in *Proc. IEEE Signal Processing Society Workshop on Multimedia Signal Processing*, December 1998, pp. 529–534.

[86] ——, "High quality scalable stereo audio coding," 1999, http://www.cipr.rpi.edu/~pearlman/papers/scal_audio.ps.gz (Data last viewed 13. January 2010). [Online]. Available: http://www.cipr.rpi.edu/~pearlman/papers/scal_audio.ps.gz

[87] N. Ma, P. Green, and A. Coy, "Exploiting Dendritic Autocorrelogram Structure to Identify Spectro-Temporal Regions Dominated by a Single Sound Source," *Speech Communication*, vol. 49, pp. 874–891, 2007.

[88] S. Mallat and Z. Zhang, "The matching pursuit software package (mpp)," data last viewed 23. April 2009. [Online]. Available: ftp://cs.nyu.edu/pub/wave/software/mpp.tar.Z

[89] ——, "Matching pursuit in a time-frequency dictionary," *IEEE Transactions on Signal Processing*, vol. 41, no. 12, pp. 3397–3415, 1993.

[90] H. Malvar, "A modulated complex lapped transform and its applications to audioprocessing," *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP99)*, 1999.

[91] G. A. Manley, "Cochlear mechanisms from a phylogenetic viewpoint," *Proceedings of the National Academy of Sciences*, vol. 97, no. 22, pp. 11 736–11 743, October 2000.

[92] D. Miyaguchi, "Abc/hidden reference audio comparison tool (version 1.0)," http://ff123.net/abchr/abchr.html, May 2004, data last viewed 13. Januar 2010. [Online]. Available: http://ff123.net/abchr/abchr.html

[93] B. C. J. Moore, *Cochlear hearing loss*. Malden, USA: Wiley-Interscience, 1998.

[94] B. Moore, *An introduction to the psychology of hearing*, 5th ed. Academic press, 2003.

[95] B. Moore and B. Glasberg, "A Revision of Zwicker's Loudness Model," *Acta Acustica united with Acustica*, vol. 82, no. 2, pp. 335–345, 1996.

[96] B. Moore, R. Peters, and B. Glasberg, "Auditory filter shapes at low center frequencies," *J. Acoust. Soc. Am*, vol. 88, pp. 132–140, 1990.

[97] J. Morlet, G. Arens, I. Fourgeau, and D. Giard, "Wave propagation and sampling theory," *Geophysics*, vol. 47, no. 2, pp. 203–236, 1982.

[98] R. Neff and A. Zakhor, "Very low bit-rate video coding based on matching pursuits," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 7, no. 1, pp. 158–171, 1997.

[99] B. Olshausen *et al.*, "Emergence of simple-cell receptive field properties by learning a sparse code for natural images," *Nature*, vol. 381, no. 6583, pp. 607–609, 1996.

[100] B. Olshausen and D. Field, "Sparse coding of sensory inputs," *Curr Opin Neurobiol*, vol. 14, no. 4, pp. 481–487, 2004.

[101] A. Oppenheim and R. Schafer, *Discrete-time signal processing.* Prentice-Hall, Inc. Upper Saddle River, NJ, USA, 1989.

[102] T. Painter and A. Spanias, "Perceptual coding of digital audio," *Proceedings of the IEEE*, vol. 88, no. 4, pp. 451–515, 2000.

[103] R. Patterson, "Auditory images: How complex sounds are represented in the auditory system," *Acoustical Science and Technology*, vol. 21, no. 4, pp. 183–190, 2000.

[104] R. Patterson and B. Moore, "Auditory filters and excitation patterns as representations of frequency resolution," in *Frequency Selectivity in Hearing*, B. Moore, Ed. London: Academic Press, 1986, pp. 123–177.

[105] R. Patterson, I. Nimmo-Smith, J. Holdsworth, and P. Rice, "An efficient auditory filterbank based on the gammatone function," *APU Report*, vol. 2341, 1988.

[106] M. Raad and A. Mertins, "From lossy to lossless audio coding using spiht," in *Proc. 5th Int. Conference on Digital Audio Effects (DAFx-02)*, Hamburg, Germany, September 2002, pp. 245–250.

[107] M. Raad, A. Mertins, and I. Burnett, "Audio coding based on the modulated lapped transform (MLT) and set partitioning in hierarchical trees," in *Proc. 6th World Multiconference on Systemics, Cybernetics and Informatics (SCI 2002)*, vol. 3, Orlando, USA, July 2002, pp. 303–306.

[108] M. Raad, A. Mertins, and I. S. Burnett, "Scalable to lossless audio compression based on perceptual set partitioning in hierarchical trees (PSPIHT)," in *Proc. Fourth International Conference on Multimedia and Expo (ICME 2003), reprinted from ICASSP'03, Baltimore, USA*, vol. 3, July 2003, pp. 393–396.

[109] S. A. Ramprashad, "High quality embedded wideband speech coding using an inherently layered coding paradigm," in *Proc. Int. Conf. on Acoustics, Speech, and Signal Processing (ICASSP'00)*, Istanbul, Turkey, June 2000, pp. 1145–1148.

[110] S. Rosen and R. J. Baker, "Characterising auditory filter nonlinearity," *Hearing Research*, vol. 73, p. 231243., 1994.

[111] A. Said and W. A. Pearlman, "A new, fast and efficient image codec based on set partitioning in hierarchical trees," *IEEE Trans. on Circuits and Systems for Video Technology*, vol. 6, no. 3, pp. 243–250, 1996.

[112] N. Schijndel, J. Bensa, M. Christensen, C. Colomes, B. Edler, R. Heusdens, J. Jensen, S. Jensen, W. Kleijn, V. Kot *et al.*, "Adaptive RD Optimized Hybrid Sound Coding," *J. Audio Eng. Soc.*, vol. 56, no. 10, pp. 787–809, 2008.

[113] J. M. Shapiro, "Embedded image coding using zerotrees of wavelet coefficients," *IEEE Trans. on Signal Processing*, vol. 41, no. 12, pp. 3445–3462, 1993.

[114] E. P. Simoncelli, "Statistical models for images: compression, restoration and synthesis," in *Conf. Record of the 35th Asilomar Conf. on Signals, Systems and Computers*, Pacific Grove, USA, November 1997, pp. 673–678.

[115] E. Smith and M. Lewicki, "Efficient Coding of Time-Relative Structure Using Spikes," *Neural Computation*, vol. 17, pp. 19–45, 2005.

[116] E. Smith, "Efficient auditory coding," Ph.D. dissertation, Carnegie Mellon University, 2006.

[117] E. Smith and M. Lewicki, "Efficient auditory coding," *Nature*, vol. 439, no. 7079, pp. 978–982, 2006.

[118] L. Solbach, R. Wöhrmann, and J. Kliewer, "The complex-valued continuous wavelet transform as a preprocessor for auditory scene analysis," in *Computational auditory scene analysis*, H. G. O. David F. Rosenthal, Ed. Lawrence Erlbaum Associates, 1998, pp. 273–291.

[119] S. Strahl and A. Mertins, "Sparse gammatone signal model optimized for English speech does not match the human auditory filters," *Brain Research*, vol. 1220, pp. 224–233, 2008.

[120] S. Strahl, H. Zhou, and A. Mertins, "An adaptive tree-based progressive audio compression scheme," in *IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA'05)*, New Paltz, USA, 2005, pp. 219–222.

[121] R. D. P. Toshio Irino, "Dynamic, Compressive Gammachirp Auditory Filterbank for Perceptual Signal Processing," in *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP06)*, 2006, pp. 133–136.

[122] P. Vaidyanathan, *Multirate systems and filter banks*. Prentice-Hall, Inc. Upper Saddle River, NJ, USA, 1993.

[123] L. Van Immerseel and S. Peeters, "Digital implementation of linear gammatone filters: Comparison of design methods," *Acoustics Research Letters Online*, vol. 4, p. 59, 2003.

[124] A. Wright, A. Davis, G. Bredberg, L. Ulehlova, and H. Spencer, "Hair cell distributions in the normal human cochlea." *Acta oto-laryngologica. Supplementum*, vol. 444, p. 1, 1987.

[125] R. Yu, R. Geiger, S. Rahardja, J. Herre, X. Lin, and H. Huang, "Mpeg-4 scalable to lossless audio coding," in *Proc. AES 117th Convention*, San Francisco, USA, October 2004, preprint 6183.

[126] L. Zadeh, "Frequency Analysis of Variable Networks," *Proceedings of the IRE*, vol. 38:3, no. 32, pp. 291–299, March 1950.

[127] H. Zhou, A. Mertins, and S. Strahl, "An efficient, fine-grain scalable audio compression scheme," in *Proc. AES 118th Convention*, Barcelona, Spain, May 2005, preprint 6435.

[128] E. Zwicker, "Subdivision of the Audible Frequency Range into Critical Bands (Frequenzgruppen)," *The Journal of the Acoustical Society of America*, vol. 33, p. 248, 1961.

Danksagungen

Diese Arbeit entstand während meiner Zeit als Mitglied im internationalen Graduiertenkolleg “Neurosensory Science and Systems” an der Carl-von-Ossietzky-Universität, Oldenburg sowie im Rahmen des Sonderforschungsbereiches “Das Aktive Gehör” (SFB/TRR31). An dieser Stelle möchte ich mich bei den vielen Personen bedanken, die mich auf dem Weg der Promotion begleitet und unterstützt haben.

Mein erster Dank gilt Prof. Dr.-Ing. Alfred Mertins für die Betreuung dieser Doktorarbeit. Er hat mit seinen überabzählbar vielen Ideen diese Arbeit immens bereichert und sehr unterstützt. Desweiteren möchte ich mich bei PD Dr. Volker Hohmann bedanken, der das Zweitgutachten übernommen hat und ebenfalls mit vielen Ideen und Denkansätzen meine Arbeit unterstützt hat. Prof. Dr. Georg Klump danke ich für die Übernahme des Beisitzes im Disputationsausschuss sowie für seine Unterstützung und seinen Rat in wissenschaftlichen sowie beruflichen Fragen. Auch möchte ich mich bei Prof. Dr. Dr. Birger Kollmeier für seine Unterstützung und seinen Rat bedanken.

Besonders möchte ich mich bei Dr. Rainer Beutelmann und Dr. Markus Kallinger für die vielen spannenden Diskussionen und vor allem die Freundschaft bedanken. Mein Dank gilt auch der Arbeitsgruppe Medizinische Physik, in der ich bei Fragen auf viele offene Ohren getroffen bin. Ich denke gerne an die vielen MEDI-Teeküchen Diskussionen zurück, und möchte besonders Dr. Jörn Arnemüller, Dr. Helmut Riedel, Dr. Rainer Huber, Dr. Thomas Rohdenburg und Prof. Jesko Verhey für die vielen hilfreichen Diskussionen danken. Vermissen werde ich auch meine vielen InterGK Mitstipendiaten und Freunde, insbesondere Dr. Julia Maier-McAlpine, Dr. Rike Steenken, Dr. Melanie Zokoll-van-der-Lahn, Dr. Melina Brell, Dr. Michael Buschemöhle, Dr. Stephan Heise und natürlich meine Mitinsassen im Graduiertenghetto, Dr. Suzan Emiroğlu, Dr. Marc Nitschmann und Dr. Gerke Hoiting sowie meinen Zimmerkollegen Dr. Heiko Hansen. Und natürlich alle Signalcos, insbesondere Dr. Radek Mazur und Jan Rademacher. Auch möchte ich mich für die logistische Hilfe und Unterstützung bei Susanne Garre, Anita Gorges, Frank Grunau und Ingrid Wusowski bedanken.

Diese Arbeit ist auch während meiner Zeit als PräPostDoc ;) am UCL Ear Institute, London enstanden und ich möchte mich für die vielen Diskussionen und die Unterstützung bei Prof. David McAlpine, Prof. Alf Linney und Torsten Marquardt bedanken.

Ein ganz besonderer Dank gilt insbesondere auch meinen Eltern für ihre große Unterstützung auf meinem Weg bis zur Promotion.

Am Schluss möchte ich mich bei der wichtigsten Person in meinem Leben bedanken, ohne die es diese Arbeit in dieser Form nie gegeben hätte. Danke Astrid für Deine seelische Unterstützung, fachliche Diskussion, unermüdliche Manuskriptkorrektur, tiefe Freundschaft und unendliche Liebe! Ich widme Dir diese Doktorarbeit und frage Dich auf diesem Wege, ob Du mich heiraten möchtest! :*

Lebenslauf

Dipl.-Math. M.Sc. Stefan Strahl

geboren am 6. März 1975
in Augsburg

Staatsangehörigkeit: deutsch

September 2003 bis April 2009	Promotion in der Arbeitsgruppe “Signalverarbeitung”, Carl-von-Ossietzky Universität Oldenburg
	Stipendiat des Graduiertenkollegs “Neurosensorik” bis August 2006, danach assoziiertes Mitglied
Juli 2003	Diplom Mathematik Titel der Diplomarbeit: “3D-Extraktion und fraktale Interpolation der menschlichen Kortexoberfläche”
Oktober 1995 bis Juli 2003	Diplomstudium der Mathematik mit der Studienrichtung Informatik und dem Anwendungsfach medizinische Biologie, Leibniz Universität Hannover
März 2000	Master of Science with distinction, Titel der Masterarbeit: “The Benefits of Natural Scenes in Data Visualization”
September 1998 bis September 1999	Masterstudium in Intelligent Systems, Brunel University, London, U.K.
Juli 1994 - September 1995	Zivildienst
Juni 1994	Abitur am Johannes Kepler Gymnasium, Garbsen

Erklärung

Hiermit erkläre ich, dass ich die vorliegende Dissertation selbständig verfasst und nur die angegebenen Hilfsmittel verwendet habe. Die Dissertation hat weder in Teilen noch in ihrer Gesamtheit einer anderen wissenschaftlichen Hochschule zur Begutachtung in einem Promotionsverfahren vorgelegen. Teile der Dissertation wurden bereits veröffentlicht bzw. sind zur Veröffentlichung eingereicht, wie an den entsprechenden Stellen angegeben. Der Anteil der Koautoren an den Veröffentlichungen bestand in der Betreuung der Arbeit und Korrektur der Manuskripte. Die Entwicklung, Programmierung und Evaluierung der Algorithmen lagen in meiner Hand.

Innsbruck, den 24. April 2009

.....

Stefan Strahl